About VISDOM

VISDOM was the name of a meter data research project, funded by ARPAe at Stanford. It stands for Visualization and Insight System for Demand Operations and Management and was conducted with PI prof. Ram Rajagopal, research staff Dr. June Flora and Dr. Chin-Woo Tan, and graduate students at the time, including Sam Borgeson and Jungsuk Kwac.

The VISDOM R package is an open source release of the software developed under that project. The core value of VISDOM lies in its ability to format, validate, and analyze large samples of utility interval meter data, typically one customer at a time. The software can be found at https://github.com/ConvergenceDA/visdom. A write up of the original use of the software can be found here: https://link.springer.com/chapter/10.1007/978-3-319-20889-3_32. Sam Borgeson's PhD thesis, whose analysis developed/used VISDOM, can be found here: https://escholarship.org/content/qt32q1w1sf/qt32q1w1sf.pdf.

Work on load shape clustering, implemented by Jungsuk Kwac and described in this paper https://ieeexplore.ieee.org/document/6693793/ is available as a separate R package here: https://github.com/ConvergenceDA/visdomloadshape.

A companion data visualization server written in Python and JavaScript using D3 visualization can be used to filter and plot extracted VISDOM features and is available here: https://github.com/ConvergenceDA/visdom-web.

VISDOM install

VISDOM is written as a standards conformant R package, but is not distributed through CRAN. Instead, you can use the package devtools to install and test it directly from GitHub:

```
# you must have already run
# install.packages('devtools')
devtools::install_github('convergenceda/visdom' , build_vignettes=T)
# this will install VISDOM and all its dependencies.
# R is not great at dependency management, so if you get errors about specific
# packages not being installed, call install.packages('the_package_mentioned')
# and re-run the install_github command above.

# 100 fake customers with random data - you need to implement your own data source
# make sure it passes a sanity check (and read through all the output)!
# DATA_SOURCE = YourDataSource()
# sanityCheckDataSource(DATA_SOURCE)
DATA_SOURCE = visdom::TestData(n=100)
visdom::sanityCheckDataSource(DATA_SOURCE)
run_results = visdom::iterator.iterateMeters( DATA_SOURCE$getIds(),
                                              visdom::basicFeatures, as_df=T )
head(run_results)
visdom::plot.MeterDataClass(DATA_SOURCE$getMeterDataClass( DATA_SOURCE$getIds()[1] ))
```

VISDOM usage overview

VISDOM is an open source code base in R that provides two important/relevant things:

- It requires conformance with a set of specific data formatting conventions for meter, customer/account, and weather data. A VISDOM "data source" retrieves data and formats it in the required format(s). The first thing you do to work with your data in VISDOM is develop a working data source for it. For some applications, just the data source support is required and you can branch off and do further analysis on your own. For more information, see:
    - https://github.com/ConvergenceDA/visdom/blob/master/vignettes/authoring_data_source.rmd, acccessible in R via `vignette('authoring_data_source', package='visdom')`.
    - `?visdom::DataSource` for information on the default DataSource skeleton function that most DataSources are built from.
    - `?visdom::TestData` for information on a complete DataSource, instantiated as `DATA_SOURCE = TestData(n=100)` that provides properly formatted but randomly generated data for testing and examples.
    - https://github.com/ConvergenceDA/visdom/blob/master/install/generic_visdom_data_source.R for the implementation of a generic data base backed DataSource, with tables defined via https://github.com/ConvergenceDA/visdom/blob/master/install/generic_create.sql and hypothetical data imports from csvs conducted via https://github.com/ConvergenceDA/visdom/blob/master/install/generic_insert.sql
    - `?visdom::sanityCheckDataSource` for information on the automatic data checking function that can help you ensure that your data source meets the needs of the rest of VISDOM.
- For data in the formats required, it provides a library of validation, analysis, and visualization tools all geared towards extracting and understanding individual and group consumption characteristics. For batch analysis of individual customers, there are many options in VISDOM's "iterator" framework to loop over customers, run standard or customer feature extraction functions, and aggregate the results into useful formats for further processing. See:
    - `?visdom::basicFeatures` for an example of a feature extraction function
    - `?visdom::iterator.iterateMeters` for an example of a function that loops over a list of customer ids running feature extraction functions on each.
    - `?visdom::iterator.iterateZip` for an example that more efficiently loops over all customers from varying zip code geographies by loading the relevant weather data only once per-location.
    - `?visdom::iterator.todf` a utility function that flattens the list of feature lists returned by the iterator functions to a data.frame of scalar values.
    - `?visdom::iterator.runMeter` a useful function that runs and returns feature extraction functions for a single customer id – primarily for testing feature functions.
    - `vignette('example_feature_extraction', package='visdom')` for a more detailed write up of all the internals of feature extraction, or viewable in version control here https://github.com/ConvergenceDA/visdom/blob/master/vignettes/example_feature_extraction.rmd.

VISDOM basic features

The function `visdom::basicFeatures` (implemented here https://github.com/ConvergenceDA/visdom/blob/master/R/features-basic.R) extracts dozens of features from meter/weather data and assigns them short names that become the column names when features are summarized as a data frame. A description of the categories the features can be broken down into is available here: https://github.com/ConvergenceDA/visdom/blob/master/VISDOM_feature_categories.pdf. A more detailed summary of variable names, units, and human readable labels is available as a CSV here: https://github.com/ConvergenceDA/visdom-web/blob/master/data/example_META.csv.