Python Data Audit Library API

Release 1.00

Wenqiang Feng and Ming Chen

April 27, 2019

CONTENTS

Welcome to our **Python Data Audit Library API**! The PDF version can be downloaded from HERE.

CHAPTER

ONE

PREFACE

Chinese proverb

Good tools are prerequisite to the successful execution of a job. – old Chinese proverb

1.1 About

1.1.1 About this API

This document is the `API` for Our Python Data Audit Library [PyAudit] API. The PDF version can be downloaded from HERE. **You may download and distribute it. Please be aware, however, that the note contains typos as well as inaccurate or incorrect description.**

In this repository, I try to use the detailed demo code and examples to show how to use `Sphinx` to generate the `.html` and `.pdf` documents and how to hookup them automatically on `Github`. If you find your work wasn't cited in this note, please feel free to let me know.

Although I am by no means a python programming and Sphinx expert, I decided that it would be useful for me to share what I learned about Sphinx in the form of easy tutorials with detailed example. I hope those tutorials will be a valuable tool for your studies.

The tutorials assume that the reader has a preliminary knowledge of `python` programing, `LaTex` and `Linux`. And this document is generated automatically by using sphinx.

1.1.2 About the author

- **Wenqiang Feng**
  - Sr. Data Scientist and PhD in Mathematics
  - University of Tennessee at Knoxville

  - Webpage: http://web.utk.edu/~wfeng1/
  - Email: von198@gmail.com

- **Ming Chen**
  - Data Scientist and PhD in Genome Science and Technology
  - University of Tennessee at Knoxville
  - Email: ming.chen0919@gmail.com

- **Biography**

  Wenqiang Feng is Data Scientist within DST's Applied Analytics Group. Dr. Feng's responsibilities include providing DST clients with access to cutting-edge skills and technologies, including Big Data analytic solutions, advanced analytic and data enhancement techniques and modeling.

  Dr. Feng has deep analytic expertise in data mining, analytic systems, machine learning algorithms, business intelligence, and applying Big Data tools to strategically solve industry problems in a cross-functional business. Before joining DST, Dr. Feng was an IMA Data Science Fellow at The Institute for Mathematics and its Applications (IMA) at the University of Minnesota. While there, he helped startup companies make marketing decisions based on deep predictive analytics.

  Dr. Feng graduated from University of Tennessee, Knoxville, with Ph.D. in Computational Mathematics and Master's degree in Statistics. He also holds Master's degree in Computational Mathematics from Missouri University of Science and Technology (MST) and Master's degree in Applied Mathematics from the University of Science and Technology of China (USTC).

- **Declaration**

  The work of Wenqiang Feng was supported by the IMA, while working at IMA. However, any opinion, finding, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the IMA, UTK and DST.

1.2 Feedback and suggestions

Your comments and suggestions are highly appreciated. I am more than happy to receive corrections, suggestions or feedbacks through email (Wenqiang Feng: von198@gmail.com) for improvements.

CHAPTER

TWO

HOW TO INSTALL

2.1 Clone the Repository

```
git clone https://github.com/runawayhorse001/PyAudit.git
```

2.2 Install

```
cd PyAudit
pip install -r requirements.txt
python setup.py install
```

2.3 Uninstall

```
pip uninstall statspy
```

2.4 Test

```
cd PyAudit/test
python test1.py
```

test1.py

```
from PyAudit.basics        import missing_rate, zero_rate, dtypes_class

import pandas as pd

```

(continues on next page)

(continued from previous page)

```
d = {'A': [1, 0, None, 3],
     'B': [1, 0, 0, 0],
     'C': ['a', None, 'c', 'd']}

# create DataFrame
df = pd.DataFrame(d)
print(missing_rate(df))
print(zero_rate(df))


# read df
df = pd.read_csv('Heart.csv', dtype={'Sex': bool})
print(df.head(5))
(num_fields, cat_fields, bool_fields, data_types) = dtypes_class(df)

print(num_fields)
print(cat_fields)
print(bool_fields)
print(data_types)
#print(missing_rate(df))
#print(zero_rate(df))
```

Results:

```
[-1.27920153  0.84000173  1.75114469 -0.02731652 -0.56417185 -0.
↪61239996
 -1.47376967  1.39551562 -0.8559779   0.60139758]

    ------------------------------------------------------------------
↪-------------
    #       One Sample t-test
    # data:  ['y']
    # t = 3.872983346207417, df = 3, p-value = 0.030466291662170977
    # alternative hypothesis: true mean is not equal to 0.0
    # 95.0 percent confidence interval:
    # 0.4457397432391206, 4.554260256760879
    # mean of x
    #         2.5
    ------------------------------------------------------------------
↪-------------
```

CHAPTER

THREE

PYTHON DATA AUDIT FUNCTIONS

3.1 dtypes_class

PyAudit.basics.**dtypes_class**(*df_in*)

numerical, categorical and bool name list in the DataFrame

Parameters **df_in** – input pandas DataFrame

Returns numerical, categorical and bool name list

Author Wenqiang Feng and Ming Chen

Email von198@gmail.com

3.2 missing_rate

PyAudit.basics.**missing_rate**(*df_in*)

calculate missing rate for each feature in the DataFrame

Parameters **df_in** – input pandas DataFrame

Returns missing rate

Author Wenqiang Feng and Ming Chen

Email von198@gmail.com

3.3 zero_rate

PyAudit.basics.**zero_rate**(*df_in*)

calculate the percentage of 0 value for each feature in the DataFrame

Parameters **df_in** – input pandas DataFrame

Returns zero rate

Author Wenqiang Feng and Ming Chen

Email von198@gmail.com

CHAPTER

FOUR

HYPOTHESIS TESTING BASICS

4.1 t_test

statspy.tests.**t_test**(*x*, *y=None*, *mu=0.0*, *conf_level=0.95*)

Performs one and two sample t-tests on vectors of data.

same functions as t.test in r: `t.test(x, ...)`

```
t.test(x, y = NULL,
    alternative = c("two.sided", "less", "greater"),
    mu = 0, paired = FALSE, var.equal = FALSE,
    conf.level = 0.95, ...)
```

Parameters

- **x** – a (non-empty) numeric vector of data values.
- **y** – an optional (non-empty) numeric vector of data values.
- **mu** – vector of standard deviations.
- **conf_level** – confidence level of the interval.

Returns the vector of the random numbers.

Author Wenqiang Feng

Email von198@gmail.com

CHAPTER

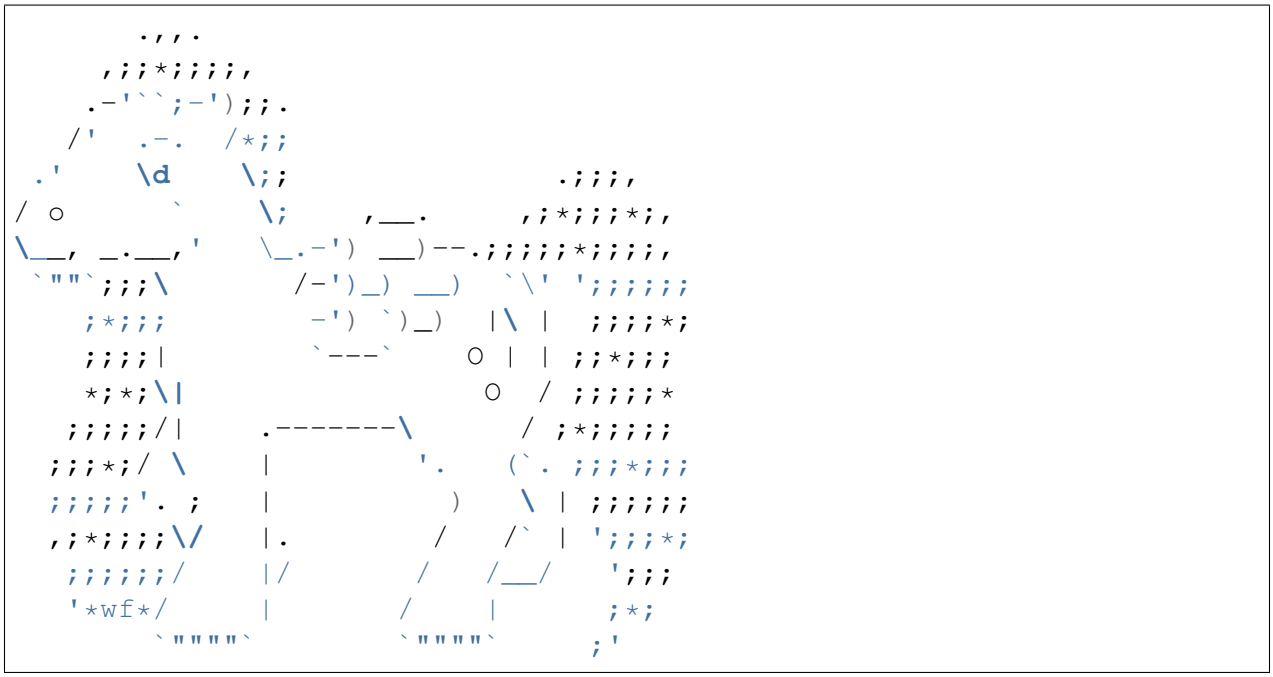
FIVE

DEMOS

This is a usage of `statspy.basics.rnorm()`:

For example:

```
>>> from statspy.basics import rnorm
>>> n=10
>>> rnorm(n)
array([ 1.54900276, -0.43444174, -0.44135064,  0.61153345, -0.31411333,
       -0.17855692, -0.35912669, -0.1131763 ,  1.64094882, -1.
↪66553673])
```



CHAPTER

SIX

MAIN REFERENCE

BIBLIOGRAPHY

[PyAudit] Wenqiang Feng and Ming Chen. Python Data Audit Library API, 2019.

PYTHON MODULE INDEX

p

s

INDEX