
PySpark Data Audit



PySparkAudit: PySpark Data Audit

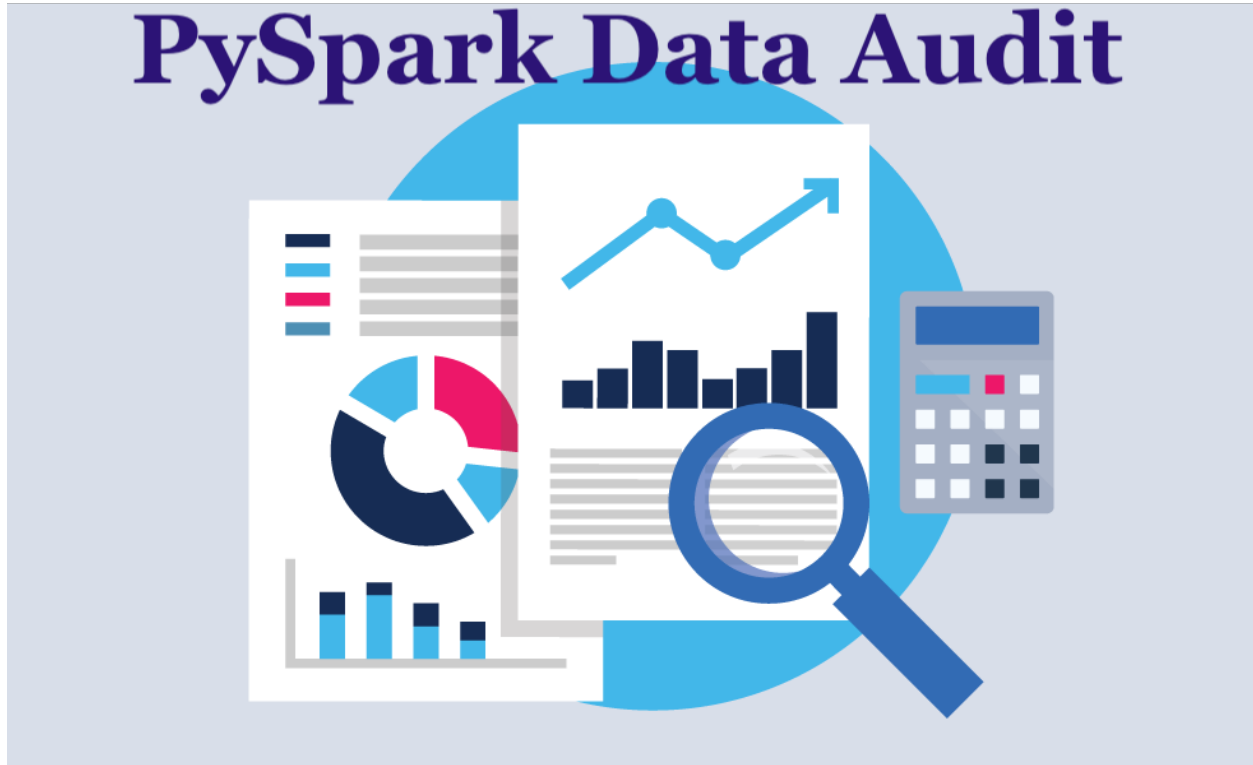
Wenqiang Feng and Yiming Xu

July 01, 2019

CONTENTS

| | | |
|----------|---|----------|
| 1 | Preface | 3 |
| 1.1 | About | 3 |
| 1.1.1 | About this API | 3 |
| 1.1.2 | About the author | 3 |
| 1.2 | Acknowledgement | 4 |
| 1.3 | Feedback and suggestions | 4 |
| 2 | How to Install | 5 |
| 2.1 | Install with <code>pip</code> | 5 |
| 2.2 | Install from Repo | 5 |
| 2.2.1 | Clone the Repository | 5 |
| 2.2.2 | Install | 5 |
| 2.2.3 | Uninstall | 5 |
| 2.2.4 | Test | 6 |
| 2.2.5 | Audited Results | 6 |
| 3 | PySpark Data Audit Functions | 9 |
| 3.1 | Basic Functions | 9 |
| 3.1.1 | <code>mkdir</code> | 9 |
| 3.1.2 | <code>mkdir_clean</code> | 9 |
| 3.1.3 | <code>df_merge</code> | 9 |
| 3.1.4 | <code>data_types</code> | 10 |
| 3.1.5 | <code>dtypes_class</code> | 10 |
| 3.1.6 | <code>counts</code> | 10 |
| 3.1.7 | <code>describe</code> | 10 |
| 3.1.8 | <code>percentiles</code> | 11 |
| 3.1.9 | <code>feature_len</code> | 11 |
| 3.1.10 | <code>freq_items</code> | 11 |
| 3.1.11 | <code>rates</code> | 12 |
| 3.1.12 | <code>corr_matrix</code> | 12 |
| 3.2 | Plot Functions | 13 |
| 3.2.1 | <code>hist_plot</code> | 13 |
| 3.2.2 | <code>bar_plot</code> | 13 |

| | | |
|----------|---|-----------|
| 3.2.3 | trend_plot | 14 |
| 3.3 | Summary Functions | 14 |
| 3.3.1 | dataset_summary | 14 |
| 3.3.2 | numeric_summary | 15 |
| 3.3.3 | category_summary | 15 |
| 3.4 | Auditing Function | 16 |
| 3.4.1 | auditing | 16 |
| 3.5 | Plotting Function | 17 |
| 3.5.1 | fig_plots | 17 |
| 4 | Auditing Demos | 19 |
| 4.1 | Auditing function by function | 19 |
| 4.2 | Auditing in one function | 21 |
| 4.3 | Auditing function by function | 23 |
| 4.3.1 | print in bash | 24 |
| 4.3.2 | Audited results folder | 26 |
| 5 | Main Reference | 33 |
| | Bibliography | 35 |



Welcome to our **PySparkAudit: PySpark Data Audit Library API**! The PDF version can be downloaded from [HERE](#).

You can install the PySparkAudit from [PyPI](<https://pypi.org/project/PySparkAudit>):

```
pip install PySparkAudit
```


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 book for our **PySparkAudit**: PySpark Data Audit Library [PySparkAudit] 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.**

The API assumes that the reader has a preliminary knowledge of python programing and Linux. And this document is generated automatically by using [sphinx](#).

The python version **PyAudit**: Python Data Audit Library API can be found at [PyAudit].

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
- **Yiming Xu**

- Data Scientist and Master of Data Science
- Harvard University
- Email: yimingxu@g.harvard.edu

- **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, DST and Harvard University.

1.2 Acknowledgement

At here, Wenqiang Feng would like to thank **Weiyu Wang** at Missouri University of Science and Technology and **Jiangtao (Lotto) Xie** at Purdue University for the unit testing and valuable discussion.

1.3 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 and Yiming Xu: yimingxu@g.harvard.edu) for improvements.

HOW TO INSTALL

2.1 Install with `pip`

You can install the `PySparkAudit` from [PyPI](<https://pypi.org/project/PySparkAudit>):

```
pip install PySparkAudit
```

2.2 Install from Repo

2.2.1 Clone the Repository

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

2.2.2 Install

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

2.2.3 Uninstall

```
pip uninstall statspy
```

2.2.4 Test

```
cd PySparkAudit/test
python test.py
```

test.py

```
from pyspark.sql import SparkSession

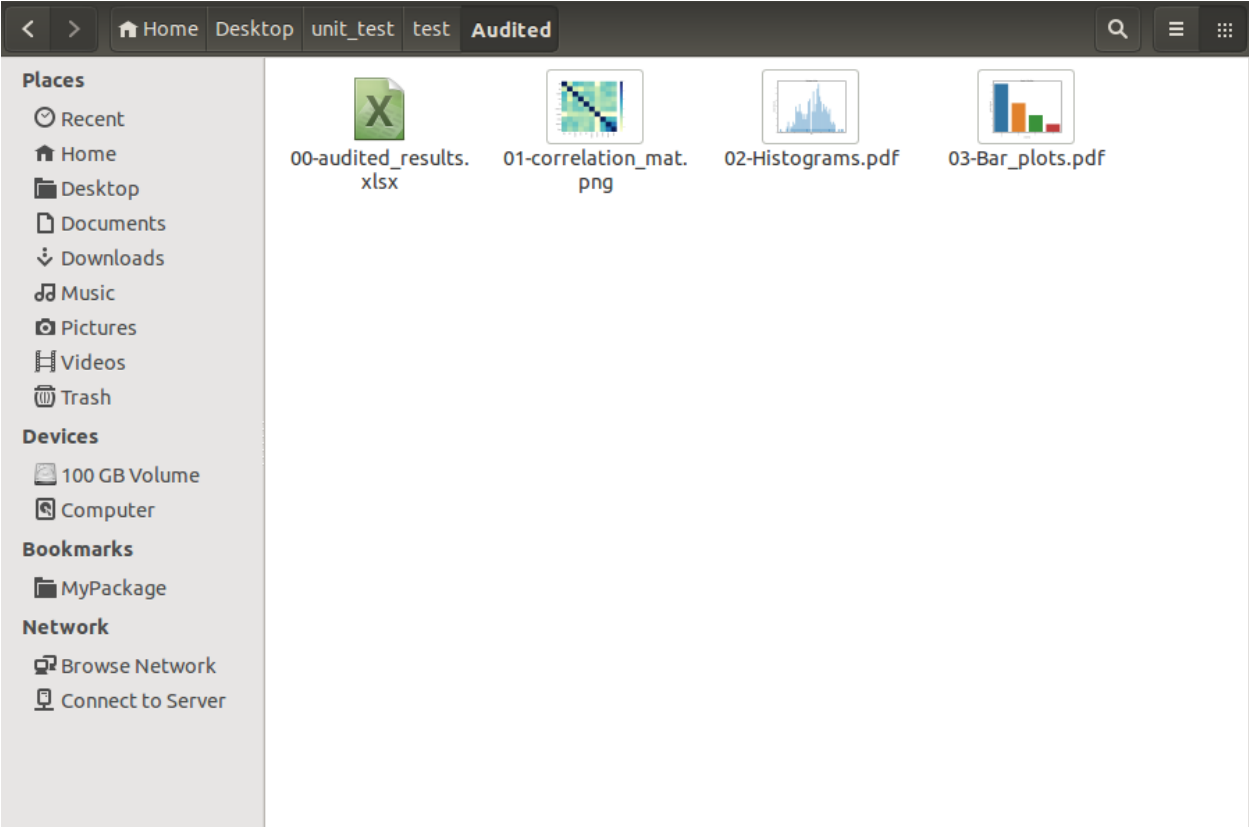
spark = SparkSession \
    .builder \
    .appName("Python Spark regression example") \
    .config("spark.some.config.option", "some-value") \
    .getOrCreate()

from PySparkAudit import dtypes_class, hist_plot, bar_plot, freq_items,
    feature_len
from PySparkAudit import dataset_summary, rates
from PySparkAudit import trend_plot, auditing
# path = '/home/feng/Desktop'

data = spark.read.csv(path='Heart.csv',
                      sep=',', encoding='UTF-8', comment=None,
    header=True, inferSchema=True)

print(auditing(data, display=True))
```

2.2.5 Audited Results



PYSPARK DATA AUDIT FUNCTIONS

3.1 Basic Functions

3.1.1 mkdir

`PySparkAudit.PySparkAudit.mkdir(path)`

Make a new directory. if it's exist, keep the old files.

Parameters `path` – the directory path

3.1.2 mkdir_clean

`PySparkAudit.PySparkAudit.mkdir_clean(path)`

Make a new directory. if it's exist, remove the old files.

Parameters `path` – the directory path

3.1.3 df_merge

`PySparkAudit.PySparkAudit.df_merge(dfs, key, how='left')`

Merge multiple pandas data frames with same key.

Parameters

- **dfs** – name list of the data frames
- **key** – key for join
- **how** – method for join, the default value is left

Returns merged data frame

3.1.4 data_types

`PySparkAudit.PySparkAudit.data_types(df_in, tracking=False)`

Generate the data types of the rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **tracking** – the flag for displaying CPU time, the default value is False

Returns data types pandas data frame

3.1.5 dtypes_class

`PySparkAudit.PySparkAudit.dtypes_class(df_in)`

Generate the data type categories: numerical, categorical, date and unsupported category.

Parameters **df_in** – the input rdd data frame

Returns data type categories

3.1.6 counts

`PySparkAudit.PySparkAudit.counts(df_in, tracking=False)`

Generate the row counts and not null rows and distinct counts for each feature.

Parameters

- **df_in** – the input rdd data frame
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the counts in pandas data frame

3.1.7 describe

`PySparkAudit.PySparkAudit.describe(df_in, columns=None, tracking=False)`

Generate the simple data frame description using `.describe()` function in pyspark.

Parameters

- **df_in** – the input rdd data frame
- **columns** – the specific feature columns, the default value is None
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the description in pandas data frame

3.1.8 percentiles

`PySparkAudit.PySparkAudit.percentiles(df_in, deciles=False, tracking=False)`

Generate the percentiles for rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **deciles** – the flag for generate the deciles
- **tracking** – the flag for displaying CPU time, the default value is False

Returns percentiles in pandas data frame

3.1.9 feature_len

`PySparkAudit.PySparkAudit.feature_len(df_in, tracking=False)`

Generate feature length statistical results for each feature in the rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the feature length statistical results in pandas data frame

3.1.10 freq_items

`PySparkAudit.PySparkAudit.freq_items(df_in, top_n=5, tracking=False)`

Generate the top_n frequent items in for each feature in the rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **top_n** – the number of the most frequent item
- **tracking** – the flag for displaying CPU time, the default value is False

Returns

3.1.11 rates

`PySparkAudit.PySparkAudit.rates(df_in, columns=None, numeric=True, tracking=False)`

Generate the null, empty, negative, zero and positive value rates and feature variance for each feature in the rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **columns** – the specific feature columns, the default value is None
- **numeric** – the flag for numerical rdd data frame, the default value is True
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the null, empty, negative, zero and positive value rates and feature variance in pandas data frame

3.1.12 corr_matrix

`PySparkAudit.PySparkAudit.corr_matrix(df_in, method='pearson', output_dir=None, rotation=True, display=False, tracking=False)`

Generate the correlation matrix and heat map plot for rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **method** – the method which applied to calculate the correlation matrix: pearson or spearman. the default value is pearson
- **output_dir** – the out put directory, the default value is the current working directory
- **rotation** – the flag for rotating the xticks in the plot, the default value is True
- **display** – the flag for displaying the figures, the default value is False
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the correlation matrix in pandas data frame

3.2 Plot Functions

3.2.1 hist_plot

```
PySparkAudit.PySparkAudit.hist_plot(df_in, bins=50, output_dir=None,
                                     sample_size=None, display=False,
                                     tracking=False)
```

Histogram plot for the numerical features in the rdd data frame. **This part is super time and memory consuming.** If the data size is larger than 10,000, the histograms will be saved in .pdf format. Otherwise, the histograms will be saved in .png format in hist folder.

If your time and memory are limited, you can use sample_size to generate the subset of the data frame to generate the histograms.

Parameters

- **df_in** – the input rdd data frame
- **bins** – the number of bins for generate the bar plots
- **output_dir** – the out put directory, the default value is the current working directory
- **sample_size** – the size for generate the subset from the rdd data frame, the default value none
- **display** – the flag for displaying the figures, the default value is False
- **tracking** – the flag for displaying CPU time, the default value is False

3.2.2 bar_plot

```
PySparkAudit.PySparkAudit.bar_plot(df_in, top_n=20, rotation=True, output_dir=None, display=False, tracking=False)
```

Bar plot for the categorical features in the rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **top_n** – the number of the most frequent feature to show in the bar plot
- **rotation** – the flag for rotating the xticks in the plot, the default value is True
- **output_dir** – the out put directory, the default value is the current working directory
- **display** – the flag for displaying the figures, the default value is False

- **tracking** – the flag for displaying CPU time, the default value is False

3.2.3 trend_plot

`PySparkAudit.PySparkAudit.trend_plot(df_in, types='day', d_time=None, rotation=True, output_dir=None, display=False, tracking=False)`

Trend plot for the aggregated time series data if the rdd data frame has date features and numerical features.

Parameters

- **df_in** – the input rdd data frame
- **types** – the types for time feature aggregation: day, month, year, the default value is day
- **d_time** – the specific feature name of the date feature, the default value is the first date feature in the rdd data frame
- **rotation** – the flag for rotating the xticks in the plot, the default value is True
- **output_dir** – the out put directory, the default value is the current working directory
- **display** – the flag for displaying the figures, the default value is False
- **tracking** – the flag for displaying CPU time, the default value is False

3.3 Summary Functions

3.3.1 dataset_summary

`PySparkAudit.PySparkAudit.dataset_summary(df_in, tracking=False)`

The data set basics summary.

Parameters

- **df_in** – the input rdd data frame
- **tracking** – the flag for displaying CPU time, the default value is False

Returns data set summary in pandas data frame

3.3.2 numeric_summary

```
PySparkAudit.PySparkAudit.numeric_summary(df_in, columns=None,  
                                           deciles=False, top_n=5,  
                                           tracking=False)
```

The auditing function for numerical rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **columns** – the specific feature columns, the default value is None
- **deciles** – the flag for generate the deciles
- **top_n** – the number of the most frequent item
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the audited results for the numerical features in pandas data frame

3.3.3 category_summary

```
PySparkAudit.PySparkAudit.category_summary(df_in, columns=None,  
                                           top_n=5, tracking=False)
```

The auditing function for categorical rdd data frame.

Parameters

- **df_in** – the input rdd data frame
- **columns** – the specific feature columns, the default value is None
- **top_n** – the number of the most frequent item
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the audited results for the categorical features in pandas data frame

3.4 Auditing Function

3.4.1 auditing

```
PySparkAudit.PySparkAudit.auditing(df_in, writer=None, columns=None,  
                                     deciles=False, top_freq_item=5,  
                                     bins=50, top_cat_item=20,  
                                     method='pearson', output_dir=None,  
                                     types='day', d_time=None, ro-  
                                     tation=True, sample_size=None,  
                                     display=False, tracking=False)
```

The wrapper of auditing functions.

Parameters

- **df_in** – the input rdd data frame
- **writer** – the writer for excel output
- **columns** – the specific feature columns, the default value is None
- **deciles** – the flag for generate the deciles
- **top_freq_item** – the number of the most frequent item
- **bins** – the number of bins for generate the bar plots
- **top_cat_item** – the number of the most frequent feature to show in the bar plot
- **method** – the method which applied to calculate the correlation matrix: pearson or spearman. the default value is pearson
- **output_dir** – the out put directory, the default value is the current working directory
- **types** – the types for time feature aggregation: day, month, year, the default value is day
- **d_time** – the specific feature name of the date feature, the default value is the first date feature in the rdd data frame
- **rotation** – the flag for rotating the xticks in the plot, the default value is True
- **sample_size** – the size for generate the subset from the rdd data frame, the default value none
- **display** – the flag for displaying the figures, the default value is False
- **tracking** – the flag for displaying CPU time, the default value is False

Returns the all audited results in pandas data frame

3.5 Plotting Function

3.5.1 fig_plots

```
PySparkAudit.PySparkAudit.fig_plots(df_in, output_dir=None,  
                                       bins=50, top_n=20, types='day',  
                                       d_time=None, rotation=True, sam-  
                                       ple_size=None, display=False,  
                                       tracking=False)
```

The wrapper for the plot functions.

Parameters

- **df_in** – the input rdd data frame
- **output_dir** – the out put directory, the default value is the current working directory
- **bins** – the number of bins for generate the bar plots
- **top_n** – the number of the most frequent feature to show in the bar plot
- **types** – the types for time feature aggregation: day, month, year, the default value is day
- **d_time** – the specific feature name of the date feature, the default value is the first date feature in the rdd data frame
- **rotation** – the flag for rotating the xticks in the plot, the default value is True
- **sample_size** – the size for generate the subset from the rdd data frame, the default value none
- **display** – the flag for displaying the figures, the default value is False
- **tracking** – the flag for displaying CPU time, the default value is False

AUDITING DEMOS

The following demos are designed to show how to use PySparkAudit to audit rdd DataFrame.

4.1 Auditing function by function

For example:

```
from pyspark.sql import SparkSession

spark = SparkSession \
    .builder \
    .appName("Python Spark regression example") \
    .config("spark.some.config.option", "some-value") \
    .getOrCreate()

# import PySpark Audit functions
from PySparkAudit import data_types, hist_plot, bar_plot, freq_items,
    ↳ feature_len
from PySparkAudit import dataset_summary, rates
from PySparkAudit import trend_plot, auditing

# load dataset
data = spark.read.csv(path='Heart.csv',
                      sep=',', encoding='UTF-8', comment=None,
    ↳ header=True, inferSchema=True)

# audit function by function

# data types
print(data_types(data))

# check frequent items
```

(continues on next page)

(continued from previous page)

```
print(freq_items(data))

# bar plot for categorical features
bar_plot(data, display=True)
```

Result:

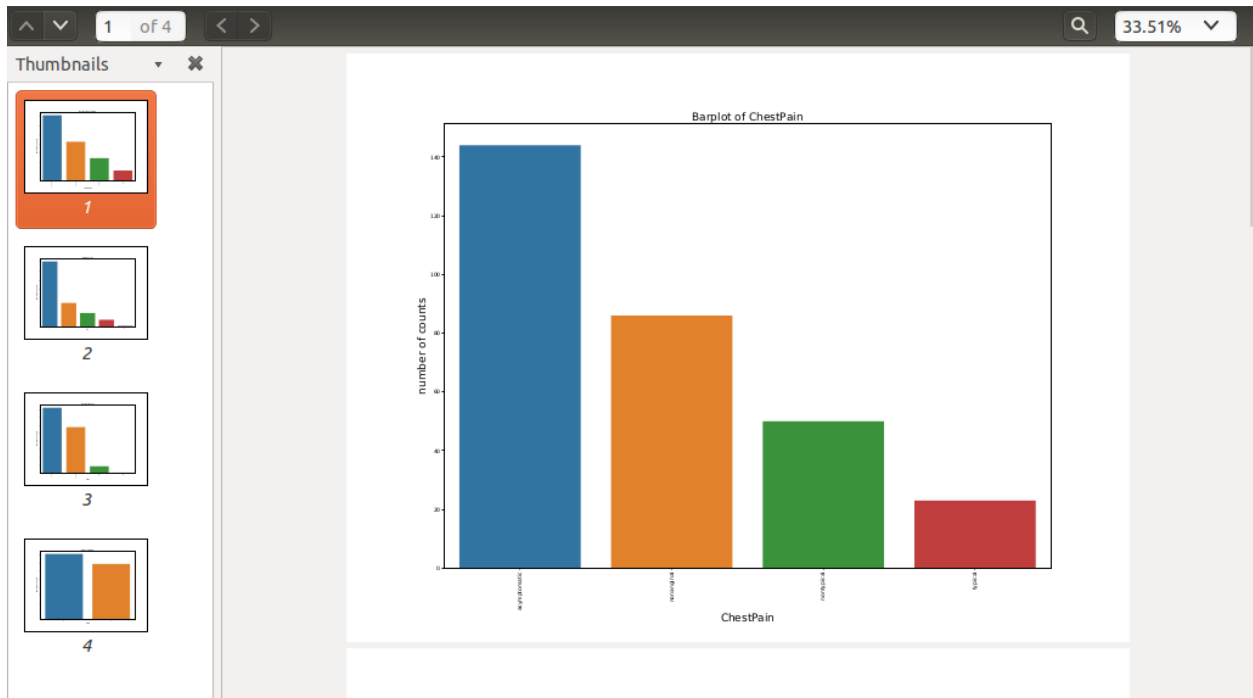
```

    feature  dtypes
0      Age      int
1      Sex      int
2  ChestPain  string
3      RestBP   int
4      Chol     int
5      Fbs      int
6  RestECG     int
7      MaxHR    int
8      ExAng    int
9  Oldpeak  double
10     Slope     int
11      Ca      string
12     Thal      string
13     AHD      string

    feature                                freq_items[value, freq]
0      Age  [[58, 19], [57, 17], [54, 16], [59, 14], [52, ...
1      Sex  [[1, 206], [0, 97]]
2  ChestPain  [[asymptomatic, 144], [nonanginal, 86], [nonty...
3      RestBP  [[120, 37], [130, 36], [140, 32], [110, 19], [...
4      Chol  [[197, 6], [234, 6], [204, 6], [254, 5], [212,...
5      Fbs  [[0, 258], [1, 45]]
6  RestECG  [[0, 151], [2, 148], [1, 4]]
7      MaxHR  [[162, 11], [163, 9], [160, 9], [152, 8], [132...
8      ExAng  [[0, 204], [1, 99]]
9  Oldpeak  [[0.0, 99], [1.2, 17], [0.6, 14], [1.0, 14], [...
10     Slope  [[1, 142], [2, 140], [3, 21]]
11      Ca  [[0, 176], [1, 65], [2, 38], [3, 20], [NA, 4]]
12     Thal  [[normal, 166], [reversable, 117], [fixed, 18]...
13     AHD  [[No, 164], [Yes, 139]]
=====
The Bar plot Bar_plots.pdf was located at:
/home/feng/Dropbox/MyTutorial/PySparkAudit/test/Audited

Process finished with exit code 0
```

and



4.2 Auditing in one function

For example:

```
from pyspark.sql import SparkSession

spark = SparkSession \
    .builder \
    .appName("Python Spark regression example") \
    .config("spark.some.config.option", "some-value") \
    .getOrCreate()

from PySparkAudit import dtypes_class, hist_plot, bar_plot, freq_items,
    feature_len
from PySparkAudit import dataset_summary, rates
from PySparkAudit import trend_plot, auditing
# path = '/home/feng/Desktop'

data = spark.read.csv(path='Heart.csv',
                      sep=',', encoding='UTF-8', comment=None,
    header=True, inferSchema=True)

print(auditing(data, display=True))
```

Result:

| | | | | | | | | | | |
|------------------------|------------|-----------|--------------|-------------------|--------------|----------|---------|-------|-----|---|
| | Age | Sex | ChestPain | RestBP | Chol | ... | Oldpeak | Slope | Ca | └ |
| → | Thal | AHD | | | | | | | | |
| 0 | 63 | True | typical | 145 | 233 | ... | 2.3 | 3 | 0.0 | └ |
| → | fixed | No | | | | | | | | |
| 1 | 67 | True | asymptomatic | 160 | 286 | ... | 1.5 | 2 | 3.0 | └ |
| → | normal | Yes | | | | | | | | |
| 2 | 67 | True | asymptomatic | 120 | 229 | ... | 2.6 | 2 | 2.0 | └ |
| → | reversible | Yes | | | | | | | | |
| 3 | 37 | True | nonanginal | 130 | 250 | ... | 3.5 | 3 | 0.0 | └ |
| → | normal | No | | | | | | | | |
| 4 | 41 | False | nontypical | 130 | 204 | ... | 1.4 | 1 | 0.0 | └ |
| → | normal | No | | | | | | | | |
| [5 rows x 14 columns] | | | | | | | | | | |
| | feature | data_type | min_digits | ... | zero_rate | pos_rate | neg_ | | | |
| → | rate | | | | | | | | | |
| Age | Age | int64 | 4 | ... | 0.000000 | 1.000000 | | | | └ |
| → | 0.0 | | | | | | | | | |
| RestBP | RestBP | int64 | 4 | ... | 0.000000 | 1.000000 | | | | └ |
| → | 0.0 | | | | | | | | | |
| Chol | Chol | int64 | 5 | ... | 0.000000 | 1.000000 | | | | └ |
| → | 0.0 | | | | | | | | | |
| Fbs | Fbs | int64 | 3 | ... | 0.851485 | 0.148515 | | | | └ |
| → | 0.0 | | | | | | | | | |
| RestECG | RestECG | int64 | 3 | ... | 0.498350 | 0.501650 | | | | └ |
| → | 0.0 | | | | | | | | | |
| MaxHR | MaxHR | int64 | 4 | ... | 0.000000 | 1.000000 | | | | └ |
| → | 0.0 | | | | | | | | | |
| ExAng | ExAng | int64 | 3 | ... | 0.673267 | 0.326733 | | | | └ |
| → | 0.0 | | | | | | | | | |
| Oldpeak | Oldpeak | float64 | 3 | ... | 0.326733 | 0.673267 | | | | └ |
| → | 0.0 | | | | | | | | | |
| Slope | Slope | int64 | 3 | ... | 0.000000 | 1.000000 | | | | └ |
| → | 0.0 | | | | | | | | | |
| Ca | Ca | float64 | 3 | ... | 0.588629 | 0.411371 | | | | └ |
| → | 0.0 | | | | | | | | | |
| [10 rows x 21 columns] | | | | | | | | | | |
| | feature | data_type | ... | top_freqs | missing_rate | | | | | |
| Sex | Sex | bool | ... | [206, 97] | 0.000000 | | | | | |
| ChestPain | ChestPain | object | ... | [144, 86, 50, 23] | 0.000000 | | | | | |
| Thal | Thal | object | ... | [166, 117, 18] | 0.006601 | | | | | |
| AHD | AHD | object | ... | [164, 139] | 0.000000 | | | | | |
| [4 rows x 10 columns] | | | | | | | | | | |

(continues on next page)

(continued from previous page)

| | Age | RestBP | Chol | ... | Oldpeak | Slope | |
|--|-----------|-----------|-----------|-----|-----------|-----------|-----|
| ↪Ca | | | | | | | |
| Age | 1.000000 | 0.284946 | 0.208950 | ... | 0.203805 | 0.161770 | 0. |
| ↪362605 | | | | | | | |
| RestBP | 0.284946 | 1.000000 | 0.130120 | ... | 0.189171 | 0.117382 | 0. |
| ↪098773 | | | | | | | |
| Chol | 0.208950 | 0.130120 | 1.000000 | ... | 0.046564 | -0.004062 | 0. |
| ↪119000 | | | | | | | |
| Fbs | 0.118530 | 0.175340 | 0.009841 | ... | 0.005747 | 0.059894 | 0. |
| ↪145478 | | | | | | | |
| RestECG | 0.148868 | 0.146560 | 0.171043 | ... | 0.114133 | 0.133946 | 0. |
| ↪128343 | | | | | | | |
| MaxHR | -0.393806 | -0.045351 | -0.003432 | ... | -0.343085 | -0.385601 | -0. |
| ↪264246 | | | | | | | |
| ExAng | 0.091661 | 0.064762 | 0.061310 | ... | 0.288223 | 0.257748 | 0. |
| ↪145570 | | | | | | | |
| Oldpeak | 0.203805 | 0.189171 | 0.046564 | ... | 1.000000 | 0.577537 | 0. |
| ↪295832 | | | | | | | |
| Slope | 0.161770 | 0.117382 | -0.004062 | ... | 0.577537 | 1.000000 | 0. |
| ↪110119 | | | | | | | |
| Ca | 0.362605 | 0.098773 | 0.119000 | ... | 0.295832 | 0.110119 | 1. |
| ↪000000 | | | | | | | |
| [10 rows x 10 columns] | | | | | | | |
| Process finished with <code>exit</code> code 0 | | | | | | | |

and

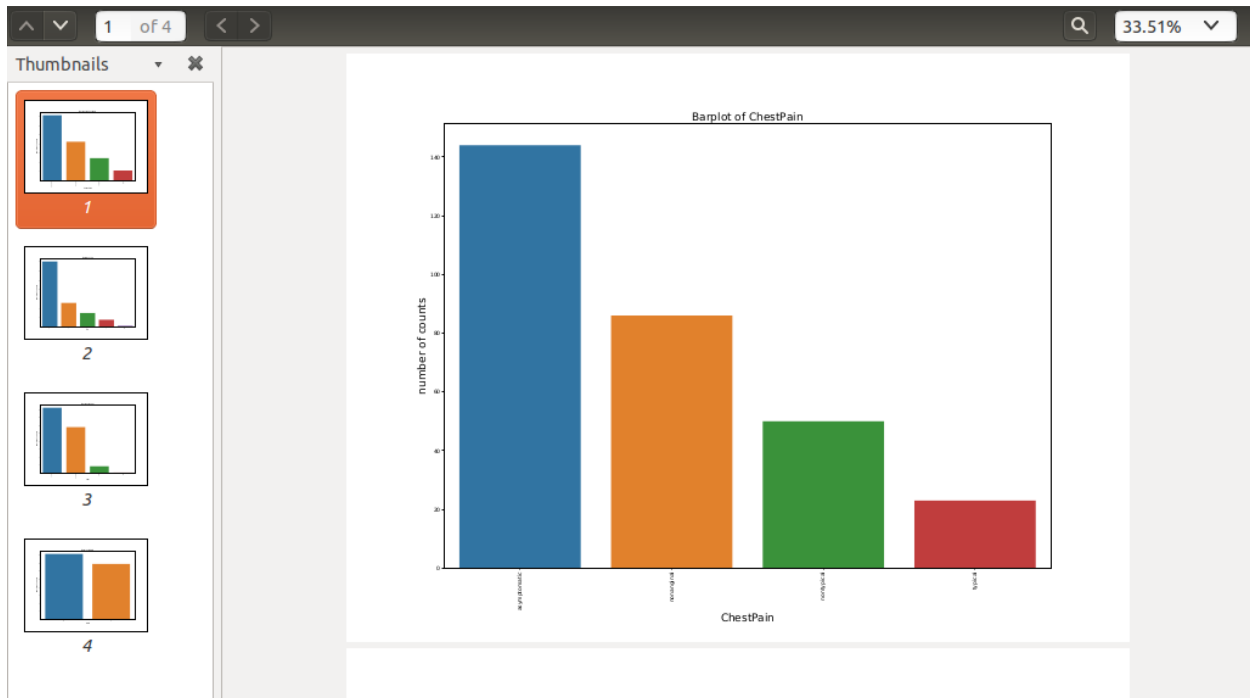
4.3 Auditing function by function

For example:

```
from pyspark.sql import SparkSession

spark = SparkSession \
    .builder \
    .appName("Python Spark regression example") \
    .config("spark.some.config.option", "some-value") \
    .getOrCreate()
```

(continues on next page)



(continued from previous page)

```
from PySparkAudit import dtypes_class, hist_plot, bar_plot, freq_items,
    feature_len
from PySparkAudit import dataset_summary, rates
from PySparkAudit import trend_plot, auditing
# path = '/home/feng/Desktop'

data = spark.read.csv(path='Heart.csv',
                      sep=',', encoding='UTF-8', comment=None,
    header=True, inferSchema=True)

print(auditing(data, display=True))
```

Result:

4.3.1 print in bash

```
=====
The audited results summary audited_results.xlsx was located at:
/home/feng/Dropbox/MyTutorial/PySparkAudit/test/Audited
=====
The correlation matrix plot Corr.png was located at:
/home/feng/Dropbox/MyTutorial/PySparkAudit/test/Audited
```

(continues on next page)

(continued from previous page)

```

=====
The Histograms plot Histograms.pdf was located at:
/home/feng/Dropbox/MyTutorial/PySparkAudit/test/Audited
Histograms plots are done!
=====

The Bar plot Bar_plots.pdf was located at:
/home/feng/Dropbox/MyTutorial/PySparkAudit/test/Audited
Caution: no date features in the dataset!!!
Generate all audited results took = 29.093122243881226 s
=====

The auditing processes are DONE!!!
(   feature  dtypes  row_count  ...  rate_neg  rate_zero  rate_pos
0      Age      int        303  ...      0.0      0.000000  1.000000
1      Sex      int        303  ...      0.0      0.320132  0.679868
2  RestBP      int        303  ...      0.0      0.000000  1.000000
3      Chol      int        303  ...      0.0      0.000000  1.000000
4      Fbs      int        303  ...      0.0      0.851485  0.148515
5  RestECG      int        303  ...      0.0      0.498350  0.501650
6    MaxHR      int        303  ...      0.0      0.000000  1.000000
7    ExAng      int        303  ...      0.0      0.673267  0.326733
8  Oldpeak  double        303  ...      0.0      0.326733  0.673267
9    Slope      int        303  ...      0.0      0.000000  1.000000

[10 rows x 22 columns],      feature  dtypes  ...  rate_null  _
→rate_empty
0  ChestPain  string      ...      0.0      0.0
1          Ca  string      ...      0.0      0.0
2        Thal  string      ...      0.0      0.0
3        AHD  string      ...      0.0      0.0

[4 rows x 12 columns],      Age      Sex      RestBP      ...  _
→    ExAng      Oldpeak      Slope
Age      1.000000 -0.097542  0.284946  ...      0.091661  0.203805  0.
→161770
Sex      -0.097542  1.000000 -0.064456  ...      0.146201  0.102173  0.
→037533
RestBP    0.284946 -0.064456  1.000000  ...      0.064762  0.189171  0.
→117382
Chol      0.208950 -0.199915  0.130120  ...      0.061310  0.046564 -0.
→004062
Fbs       0.118530  0.047862  0.175340  ...      0.025665  0.005747  0.
→059894
RestECG   0.148868  0.021647  0.146560  ...      0.084867  0.114133  0.
→133946
MaxHR     -0.393806 -0.048663 -0.045351  ...     -0.378103 -0.343085 -0.
→385601

```

(continues on next page)

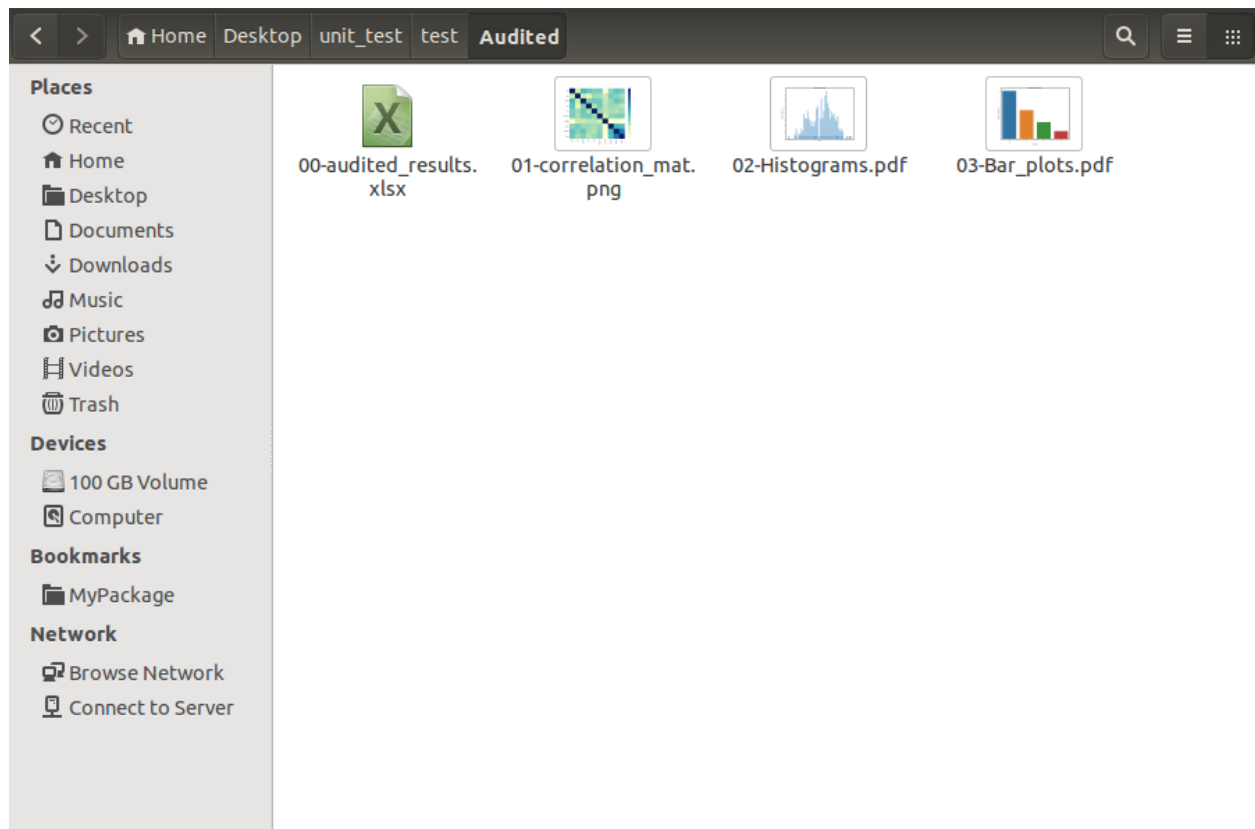
(continued from previous page)

```
ExAng      0.091661  0.146201  0.064762  ...  1.000000  0.288223  0.
↪257748
Oldpeak    0.203805  0.102173  0.189171  ...  0.288223  1.000000  0.
↪577537
Slope      0.161770  0.037533  0.117382  ...  0.257748  0.577537  1.
↪000000

[10 rows x 10 columns])

Process finished with exit code 0
```

4.3.2 Audited results folder



The files in `00-audited_results.xlsx`:

1. Dataset_summary
2. Numeric_summary

| | A | B | C | D | E | F | G | H | I | J |
|----|-----------------------|--------------|---|---|---|---|---|---|---|---|
| 1 | summary | value | | | | | | | | |
| 2 | sample_size | 303 | | | | | | | | |
| 3 | feature_size | 14 | | | | | | | | |
| 4 | single_unique_feature | 0 | | | | | | | | |
| 5 | row_w_null | 0 | | | | | | | | |
| 6 | row_w_empty | 0 | | | | | | | | |
| 7 | row_w_zero | 299 | | | | | | | | |
| 8 | row_avg_null_count | 0 | | | | | | | | |
| 9 | row_avg_empty_count | 0 | | | | | | | | |
| 10 | row_avg_zero_count | 3.25083 | | | | | | | | |
| 11 | numerical_fields | 10 | | | | | | | | |
| 12 | categorical_fields | 4 | | | | | | | | |
| 13 | date_fields | 0 | | | | | | | | |
| 14 | unsupported_fields | 0 | | | | | | | | |
| 15 | double | 1 | | | | | | | | |
| 16 | int | 9 | | | | | | | | |
| 17 | string | 4 | | | | | | | | |
| 18 | | | | | | | | | | |
| 19 | | | | | | | | | | |
| 20 | | | | | | | | | | |
| 21 | | | | | | | | | | |
| 22 | | | | | | | | | | |
| 23 | | | | | | | | | | |
| 24 | | | | | | | | | | |
| 25 | | | | | | | | | | |
| 26 | | | | | | | | | | |

Dataset_summary / Numeric_summary / Category_summary / Correlation_matrix

Find Find All Match Case

Sheet 1 / 4 PageStyle_Dataset_summary Sum=0 100%

| Calibri | | | | | | | | | | | |
|---------|---------|--------|-----------|------------|----------------|----------|----------|-----|-----|-----|-----|
| A1 | | | | | | | | | | | |
| | A | B | C | D | E | F | G | H | I | J | K |
| 1 | feature | dtypes | row_count | null_count | distinct_count | mean | stddev | min | max | Q1 | Med |
| 2 | Age | int | 303 | 303 | 41 | 54.4389 | 9.03866 | 29 | 77 | 48 | 56 |
| 3 | Sex | int | 303 | 303 | 2 | 0.679867 | 0.467298 | 0 | 1 | 0 | 1 |
| 4 | RestBP | int | 303 | 303 | 50 | 131.6897 | 17.59974 | 94 | 200 | 120 | 130 |
| 5 | Chol | int | 303 | 303 | 152 | 246.6930 | 51.77691 | 126 | 564 | 211 | 242 |
| 6 | Fbs | int | 303 | 303 | 2 | 0.148514 | 0.356197 | 0 | 1 | 0 | 0 |
| 7 | RestECG | int | 303 | 303 | 3 | 0.990099 | 0.994971 | 0 | 2 | 0 | 1 |
| 8 | MaxHR | int | 303 | 303 | 91 | 149.6072 | 22.87500 | 71 | 202 | 134 | 153 |
| 9 | ExAng | int | 303 | 303 | 2 | 0.326732 | 0.469794 | 0 | 1 | 0 | 0 |
| 10 | Oldpeak | double | 303 | 303 | 40 | 1.039603 | 1.161075 | 0.0 | 6.2 | 0 | 0.8 |
| 11 | Slope | int | 303 | 303 | 3 | 1.600660 | 0.616226 | 1 | 3 | 1 | 2 |
| 12 | | | | | | | | | | | |
| 13 | | | | | | | | | | | |
| 14 | | | | | | | | | | | |
| 15 | | | | | | | | | | | |
| 16 | | | | | | | | | | | |
| 17 | | | | | | | | | | | |
| 18 | | | | | | | | | | | |
| 19 | | | | | | | | | | | |
| 20 | | | | | | | | | | | |
| 21 | | | | | | | | | | | |
| 22 | | | | | | | | | | | |
| 23 | | | | | | | | | | | |
| 24 | | | | | | | | | | | |
| 25 | | | | | | | | | | | |
| 26 | | | | | | | | | | | |

Dataset_summary Numeric_summary Category_summary Correlation_matrix

Find Find All Match Case

Sheet 2 / 4 PageStyle_Numeric_summary Sum=0 100%

3. Category_summary

| | A | B | C | D | E | F | G | H | I | J | K |
|----|-----------|--------|-----------|------------|----------------|------------|------------|------------|-------------|----------|-----------|
| 1 | feature | dtypes | row_count | null_count | distinct_count | min_length | avg_length | max_length | ms[values] | variance | rate_null |
| 2 | ChestPain | string | 303 | 303 | 4 | 7 | 10.7228 | 12 | [[['asympt | 0.0132 | 0 |
| 3 | Ca | string | 303 | 303 | 5 | 1 | 1.0132 | 2 | [[['0', 176 | 0.0165 | 0 |
| 4 | Thal | string | 303 | 303 | 4 | 2 | 7.45875 | 10 | [[['normal | 0.0132 | 0 |
| 5 | AHD | string | 303 | 303 | 2 | 2 | 2.45875 | 3 | [[['No', 16 | 0.0066 | 0 |
| 6 | | | | | | | | | | | |
| 7 | | | | | | | | | | | |
| 8 | | | | | | | | | | | |
| 9 | | | | | | | | | | | |
| 10 | | | | | | | | | | | |
| 11 | | | | | | | | | | | |
| 12 | | | | | | | | | | | |
| 13 | | | | | | | | | | | |
| 14 | | | | | | | | | | | |
| 15 | | | | | | | | | | | |
| 16 | | | | | | | | | | | |
| 17 | | | | | | | | | | | |
| 18 | | | | | | | | | | | |
| 19 | | | | | | | | | | | |
| 20 | | | | | | | | | | | |
| 21 | | | | | | | | | | | |
| 22 | | | | | | | | | | | |
| 23 | | | | | | | | | | | |
| 24 | | | | | | | | | | | |
| 25 | | | | | | | | | | | |
| 26 | | | | | | | | | | | |

4. Correlation_matrix

Histograms in Histograms.pdf

Barplots in Bar_plots.pdf

```
....
,iii*iiii,
.-'`';-');;.
```

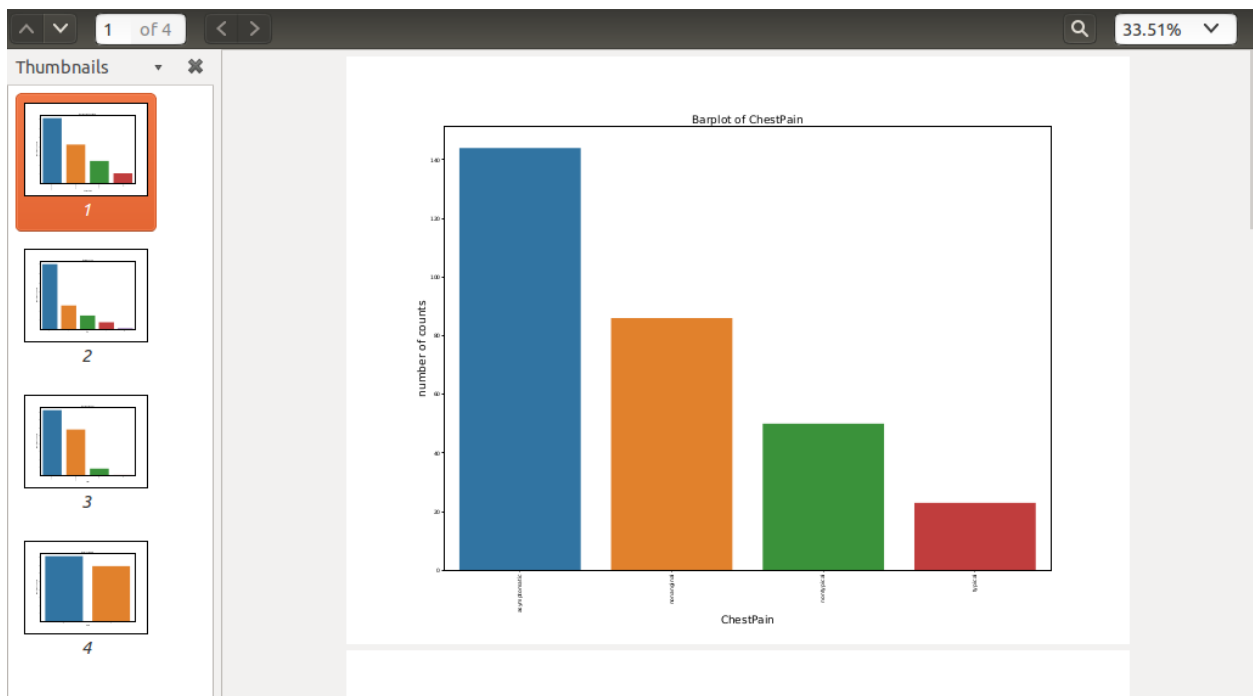
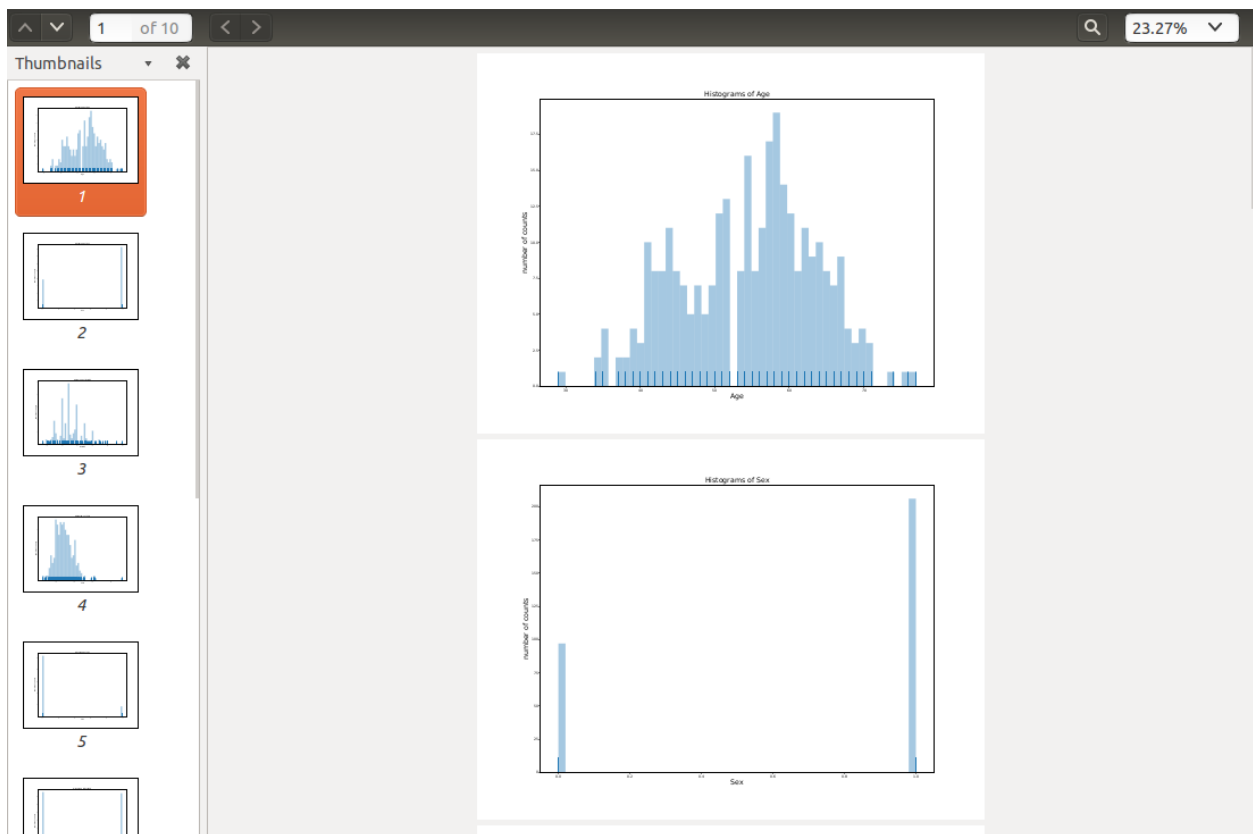
(continues on next page)

| Calibri 11 | | | | | | | | | | | |
|------------|----------|----------|----------|----------|----------|----------|----------|---------|----------|----------|--|
| A1 Age | | | | | | | | | | | |
| 1 | Age | Sex | RestBP | Chol | Fbs | RestECG | MaxHR | ExAng | Oldpeak | Slope | |
| 2 | 1 | -0.09754 | 0.28495 | 0.20895 | 0.11853 | 0.14887 | -0.39381 | 0.09166 | 0.20381 | 0.16177 | |
| 3 | -0.09754 | 1 | -0.06446 | -0.19991 | 0.04786 | 0.02165 | -0.04866 | 0.1462 | 0.10217 | 0.03753 | |
| 4 | 0.28495 | -0.06446 | 1 | 0.13012 | 0.17534 | 0.14656 | -0.04535 | 0.06476 | 0.18917 | 0.11738 | |
| 5 | 0.20895 | -0.19991 | 0.13012 | 1 | 0.00984 | 0.17104 | -0.00343 | 0.06131 | 0.04656 | -0.00406 | |
| 6 | 0.11853 | 0.04786 | 0.17534 | 0.00984 | 1 | 0.06956 | -0.00785 | 0.02567 | 0.00575 | 0.05989 | |
| 7 | 0.14887 | 0.02165 | 0.14656 | 0.17104 | 0.06956 | 1 | -0.08339 | 0.08487 | 0.11413 | 0.13395 | |
| 8 | -0.39381 | -0.04866 | -0.04535 | -0.00343 | -0.00785 | -0.08339 | 1 | -0.3781 | -0.34309 | -0.3856 | |
| 9 | 0.09166 | 0.1462 | 0.06476 | 0.06131 | 0.02567 | 0.08487 | -0.3781 | 1 | 0.28822 | 0.25775 | |
| 10 | 0.20381 | 0.10217 | 0.18917 | 0.04656 | 0.00575 | 0.11413 | -0.34309 | 0.28822 | 1 | 0.57754 | |
| 11 | 0.16177 | 0.03753 | 0.11738 | -0.00406 | 0.05989 | 0.13395 | -0.3856 | 0.25775 | 0.57754 | 1 | |
| 12 | | | | | | | | | | | |
| 13 | | | | | | | | | | | |
| 14 | | | | | | | | | | | |
| 15 | | | | | | | | | | | |
| 16 | | | | | | | | | | | |
| 17 | | | | | | | | | | | |
| 18 | | | | | | | | | | | |
| 19 | | | | | | | | | | | |
| 20 | | | | | | | | | | | |
| 21 | | | | | | | | | | | |
| 22 | | | | | | | | | | | |
| 23 | | | | | | | | | | | |
| 24 | | | | | | | | | | | |
| 25 | | | | | | | | | | | |
| 26 | | | | | | | | | | | |

Dataset_summary / Numeric_summary / Category_summary / Correlation_matrix

Find Find All Match Case

Sheet 4 / 4 PageStyle_Correlation_matrix Sum=0 100%



(continued from previous page)

```
/' .-. /*;;
.' \d \;;
/ o \; ,__ . ,; *; *;
\_, _., ' \_, -') _ ) -- .; *; *;
`""; ; \ /-') _ ) \ ' ' ; ; ;
; *; ; -') `)_ _ ) | \ | ; ; ; *;
; ; ; | `---` o | | ; ; ; *;
* ; * ; \ | o / ; ; ; *;
; ; ; ; / | .----- \ / ; ; ; ;
; ; * ; / \ | ' . ( ` . ; ; * ; ;
; ; ; ; ' . ; | ) \ | ; ; ; ;
, ; * ; ; ; \ | . / / ` | ' ; ; * ;
; ; ; ; ; / | / / _ / ' ; ;
' * w f * / | / _ | ; * ;
`""""` `""""` ; '
```

MAIN REFERENCE

BIBLIOGRAPHY

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

[PySparkAudit] Wenqiang Feng and Ming Chen. [PySpark Data Audit Library API](#), 2019.