

CSC 5741 Lecture 2: Python for Data Mining and Machine Learning



Lighton Phiri <lighton.phiri@unza.zm>
Department of Library and Information Science
University of Zambia

Announcements—March 26, 2019

- **CSC 5741 Moodle site up and running**
 - All assignments will be available and submitted via The Moodle
- **Mini Projects**
 - Open date: March 29, 2019
 - Due date: May 17, 2019
- **Paper reading #01**
 - Open date: March 29, 2019
 - Due date: April 5, 2019
 - Paper suggestion rotation—Two every week
- **Invited Academic Talk**
 - April 2, 2019—Lillian Mzyece

March 26 2019

CSC 5741 L02 - 2

Announcements—April 2, 2019 (1/3)

- No talk today—April 2, 2019—as none of the invited guests are able to show up
 - I could give another talk if people wish :P
- Mulizwa will come through to give a talk on April 9, 2019—next week
 - Biography & abstract will be sent
 - Please show up on time and ask a lot of questions

CSC 5741 Invited Talks Slots
by Lighton Phiri 1 day ago • Print

University of Zambia

All times displayed in Africa/Lusaka

| Table | | Calendar | | | | | | |
|--------------------|---|----------|-------|--------|--------|--------|-------|--------|
| | | Apr 2 | Apr 9 | Apr 16 | Apr 23 | Apr 30 | May 7 | May 14 |
| 5:00 PM | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| 6:00 PM | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Enter your name | ● | ● | ● | ● | ● | ● | ● | ● |
| Lillian Mzyece | ● | | | | | | | |
| Andrea Kumwenda | | ● | | | | | | |
| Friday C. Chisanga | | | ● | ● | | | | |
| Soft Muliwa | | ● | | | | | | |
| Francis Chulu | | | | | | ● | | |

<https://doodle.com/poll/bmv7b5yqq5nbdu9n>

CSC 5741 L02 - 3

Announcements—April 2, 2019 (2/3)

- Paper reading summaries are due on April 8, 2019
 - Please follow the reading assignment specifications

GSC 5741 - Data Mining and Warehousing :
Paper Reading Summary Assignment #01 | Open Date: April 1, 2019
1 post by 1 user(s) 0

me (Lighton Phiri) created

Write a short summary of the "Data-driven interpretable-new prediction modeling for academic performance" publication [1, 2] by Mghele and Mbogho.
* The summary should be no longer than 200 words and must be submitted as a single PDF document via The Moodle or via the course Team Drive.
* The file must be a PDF document named by STUDENTID_PAPERNAME.pdf. replace STUDENTID with your STUDENT ID.
* Use A4 page size, single line spacing, 12pt serif font and 2.54cm margins.

The deadline for submitting the summary is April 8, 2019 at 23h59 GMT+2. No late hand-ins, printed or handwritten submissions will be accepted.

[1] <https://doi.org/10.1145/273706.273802>
[2] <https://doi.org/10.1145/273706.273801>

Best wishes,

Lighton Phiri, PhD
Department of Library and Information Science
University of Zambia
Lusaka, Zambia
Email: lighton.phiri@unza.zm
Web: <http://www.unza.zm/~lightonphiri/>

Attachments | 1

42-mpga.pdf
245 KB | View | Download

<https://groups.google.com/a/unza.zm/d/forum/csc5741>

CSC 5741 L02 - 4

March 26 2019

Announcements—April 2, 2019 (3/3)

- Please make your selections as soon as possible**
 - You might want to make selections to work in packs
- Errata**
 - Distribution of marks for presentation was less than 100%
 - Page limit for technical report is four (4) pages not six (6) pages

March 26 2019

CSC 5741 Mini Project (2018/19)
by Lighton Phiri • a day ago • Print

| task | 2 (0) NETD: Classification of ETDs universities based on ETD output | 3 (0) NETD: Cluster analysis of ETDs by subject area | 3 (0) YouTube: Recommended YouTube Videos to undergraduate students | 3 (0) YouTube: Classification of random YouTube videos to undergraduate students | 3 (0) Facebook: Classification of posts on popular Zimbabwe Facebook pages | |
|------|---|--|---|--|--|-------|
| | ✓ 1/1 | ✓ 0/1 | ✓ 0/1 | ✓ 1/1 | ✓ 0/1 | ✓ 1/1 |
| | ● | ● | ● | ● | ● | ● |
| | ✓ | | | ✓ | | ✓ |
| | | | | | | |

<https://doodle.com/poll/534szgg6aw56yxtz>

CSC 5741 L02 - 5

Todos | Showcases—April 2, 2019 (1/6)

```
</metadata>
<about>
  <provenance xmlns="http://www.openarchives.org/OAI/2.0/provenance" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="oai_dc:OAIResource"
    http://www.openarchives.org/OAI/2.0/provenance.xsd">
    <originDescription harvestDate="2014-02-04T04:15:33Z" altered="false">
      <baseURL>http://etd.uovs.ac.za/cgi-bin/NDLTD0/UF5/oai.pl</baseURL>
      <identifier>oai:etd.uovs.ac.za:etd-07172013-155725</identifier>
      <datestamp>2013-07-17T04:15:33Z</datestamp>
      <metadataNamespace>http://www.openarchives.org/OAI/2.0/oai_dc</metadataNamespace>
    </originDescription>
  </provenance>
</about>
<resumptionToken completeListSize="171424" cursor="0">
  2014-02-04T16:15:33Z!2037-01-01T00:00:00Z!oai_dc!10800!171424!oai:union.ndltd.org:ufs:oai:etd.uovs.ac.za:etd-07172013-155725
</resumptionToken>
</OAI-PMH>
```

<http://www.netd.ac.za>

- Extraction of records of NETD and NDLTD Union Catalog is best done using the OAI-PMH protocol**
 - Use the 'resumptionToken' to loop through batches of records

March 26 2019

CSC 5741 L02 - 6

Todos | Showcases—April 2, 2019 (2/6)

Deprecated on Apr 19, 2019. Please use the API Upgrade Tool to understand how this might impact your app. For more details see the API Upgrade Tool documentation.

Notice: Your app is missing a Contact Email. We use this to send you important communications about your app. To learn more, see the App Configuration documentation.

New App

DISQUS Co... App ID: 322904707327 Status: Live

js-page-scr... App ID: 647142025050081 Status: In development

[+ Add a New App](#)

js-page-scr...

App ID: 647142025050081
Status: In development

Display Name: js-page-scraper

App Domain:

Contact Email:

Privacy Policy URL: Privacy policy for Login dialog and App Details

Terms of Service URL: Terms of Service for Login dialog and App Details

App Icon (1024 x 1024)

Category:

<https://developers.facebook.com/apps>

- Extraction of data from Facebook is best done using Facebook's Graph API**
 - You will need to get access tokens for this to work

March 26 2019

CSC 5741 L02 - 7

Todos | Showcases—April 2, 2019 (3/6)

[lightonphiri / code-fb-page-scrap... Private](#)

[Unwatch](#) 1 [Star](#) 0 [Fork](#) 0

[Code](#) [Issues](#) 0 [Pull requests](#) 0 [Projects](#) 0 [Wiki](#) [Insights](#) [Settings](#)

A simple JavaScript-based script for extracting information from Facebook pages.

[Edit](#)

Manage topics

7 commits 1 branch 0 releases 1 contributor

Branch: master [New pull request](#) [Create new file](#) [Upload files](#) [Find File](#) [Clone or download](#)

lightonphiri Git sync transaction

Latest commit 84390da on Jan 29, 2015

| File | Description | Age |
|------------|--------------------------------------|-------------|
| js | Git sync transaction | 4 years ago |
| .gitignore | Initial commit | 4 years ago |
| README.md | Initial commit | 4 years ago |
| index.html | Added templates and boilerplate code | 4 years ago |

<https://github.com/lightonphiri/code-fb-page-scrap...>

[March 26 2019](#)

CSC 5741 L02 - 8

Todos | Showcases—April 2, 2019 (4/6)

- YouTube data is accessible via the YouTube Data API
 - Those working with comments will need to both the 'Comments' and 'CommentThreads' resources

The screenshot shows the YouTube Data API documentation for the 'CommentThreads' resource. It includes sections for 'Overview', 'APIs', 'Performance', 'Samples', 'Support', 'Search', and 'All products'. The main content area is titled 'CommentThreads' and contains information about the resource, methods like 'list', and a code sample in JSON. At the bottom, there's a link to the API reference page.

March 26 2019

<https://developers.google.com/youtube/v3>

CSC 5741 L02 - 9

Todos | Showcases—April 2, 2019 (5/6)

- Google Scholar does not provide an API, so you are going to have to scrap for data
 - Python Beautiful Soup library might be useful
 - Some projects on GitHub and Bitbucket might be useful when extracting data from Google Scholar

The screenshot shows a GitHub repository named 'ckreibich/scholar.py'. It has 47 commits, 1 branch, and 0 releases. The repository description states it is a parser for Google Scholar, written in Python. It includes a README file and a 'scholar.py' script. The script is described as a command-line tool for extracting content excerpts from Google Scholar's output. A note at the bottom indicates the script was moved to a new location and updated.

March 26 2019

<https://github.com/ckreibich/scholar.py>

CSC 5741 L02 - 10

Todos | Showcases—April 2, 2019 (6/6)

- Read up on how to use Microsoft Academic Graph API to extract data from Microsoft Academic Search

March 26 2019

CSC 5741 L02 - 11

Announcements—April 9, 2019 (1/3)

- Paper reading suggestions account towards participation
 - Simple formula will be used to aggregate scores from other activities, e.g. talks.
- Hint: Look for papers aligned to the problem you are solving in the Mini Project

| No. | First Name | Lastname |
|-----|------------|---------------|
| 1 | Chola | Paul Modest |
| 2 | Daka | John Chrispin |
| 3 | Lamaswala | Inonge |
| 4 | Mubanga | Mubanga |
| 5 | Mukuma | Nonde |
| 6 | Mulenga | David |
| 7 | Mumbi | Memory |
| 8 | Mutende | Kaumba |
| 9 | Nongola | Justin |
| 10 | Nyambe | Teddy |
| 11 | Phiri | Jonathan |
| 12 | Sampa | Anthny Wila |
| 13 | Shamane | Tasha |

<https://groups.google.com/a/unza.zm/forum/?hl=en#!forum/csc5741>

CSC 5741 L02 - 12

March 26 2019

Announcements—April 9, 2019 (2/3)

- Mini Project questions involving Facebook's Graph API can use WordPress REST API
- Anyone managed with Graph API
 - WordPress REST API route: restrict model implementation to popular sites like LusakaTimes, ZambianWatchDog & Mwebantu.

March 26 2019

The screenshot shows the REST API Handbook interface. The left sidebar has sections for CHAPTERS, REST API Handbook, Reference, Posts, Post Revisions, Categories, Tags, Pages, Comments, Taxonomies, Media, Users, Post Types, Post Statuses, Settings, Using the REST API, and Changing. The main content area is titled 'Posts' and contains a 'TOPICS' section with links to 'Schema', 'Example Request', 'List Arguments', 'Definition', 'Create a Post', 'Edit a Post', 'Delete a Post', 'Retrieve a Post', 'Update a Post', and 'Delete a Post'. Below this is a 'Schema #' section with a detailed schema definition. At the bottom of the page is a navigation bar with links for 'View', 'Edit', and 'Deleted' and a note about the date of publication. The URL is <https://developer.wordpress.org/rest-api/>.

CSC 5741 L02 - 13

Announcements—April 9, 2019 (3/3)

```
id: 229497,
date: "2019-04-09T07:33:43",
date_gmt: "2019-04-09T05:33:43",
guid: {
  rendered: "https://www.lusakatimes.com/?p=229497"
},
modified: "2019-04-09T07:33:43",
modified_gmt: "2019-04-09T05:33:43",
slug: "faz-div-1-wrap-kansanshi-top-zone-2",
status: "publish",
type: "post",
link: "https://www.lusakatimes.com/2019/04/09/faz-div-1-wrap-kansanshi-top-zone-2/",
title: "FAZ DIV 1 WRAP: Kansanshi top Zone 2"
},
content: {
  rendered: "<p>Kansanshi Dynamos have won the first half of the 2019 FAZ Division Zone 2 season with 36 points after a 2-1 win over Zesco Luapula in the Week 15 n a three point gap half way into the season.</p> <p>The Solwezi side benefited from Zesco's own goal to triumph in their latest match at home in Solwezi.</p> <p>G home in Ndola.</p> <p>Solwezi remain stuck on 33 points from 15 matches played while Kansanshi are six points behind in third place. <p>Fourth placed Indeni and are also in a bit of a bind, having lost to Zesco Luapula 1-2 in their last 33 points from 15 matches. <p>Currently are second in the table<-> they are the top placed Kabwe Youth are three points behind.</p> <p>Zone 4 leaders Young Green Eagles have 31 points, two above second placed Zesco Shockers after 15 matches play. <p>Young Green Eagles are four points behind. <p>Zesco Shockers are five points behind. <p>Kabwe Rangers 2-2 Katue Celtic</p> <p>Paramilitary 1-1 Lusaka City Council</p> <p>National Assembly 2-1 Remeiki FC</p> <p>ZONE TWO</p> <p>Mololo United 3-1 Roan United</p> 0-0 Indeni</p> <p>Changa Rangers 2-6 FOM FM</p> <p>Jongens 0-1 Konkola Blades</p> <p>DZNS Luafumu 2-0 Trident</p> <p>Kansanshi Dynamos 1-0 Zesco Luapula</p> <p>Ka 0-0 Chilima United</p> <p>Young Green Eagles 2-1 Chilima United</p> <p>Young Green Eagles 2-0 Tazara Rangers</p> <p>Real Nakonde 3-1 Mpulungu Harbour</p> <p>Kabwe Rangers 2-0 Chindwin Sentriess</p> <p>ZONE FOUR</p> <p>Mazab Medics</p> <p>Sinazongwa United 1-0 Zesco Shockers</p> <p>Choma football Stars 0-2 Livingstone Pirates</p> <p>Salomo Jetters 1-0 New Monze Swallows</p> <p>Young Green Eagles 0-0 Arrows 0-0 Katima Border Stars</p> ",
protected: false
},
excerpt: {
  rendered: "<p>Kansanshi Dynamos have won the first half of the 2019 FAZ Division Zone 2 season with 36 points after a 2-1 win over Zesco Luapula in the Week 15 n a three point gap half way into the season. The Solwezi side benefited from Zesco's [delip].</p> ",
protected: false
}
},
link: "https://www.lusakatimes.com/wp-json/wp/v2/posts/229497"
},
March 26 2019
```

<https://www.lusakatimes.com/wp-json/wp/v2/posts/229497>

CSC 5741 L02 - 14

Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week

March 26 2019

CSC 5741 L02 - 15

Lecture Series Outline

- Part I: Getting Started With Python
 - Introduction
 - Installation and Setup
 - Basics
 - Data Structures
 - Flow Control
 - Functions and Modules
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week

March 26 2019

CSC 5741 L02 - 16

Getting Started With Python (1/3)

```
Python 3.6.7 (default, Oct 22 2018, 11:32:17)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
```

March 26 2019

CSC 5741 L02 - 17

Getting Started With Python (2/3)

- Python is an interpreted language
- Python is a scripting language
- Python is a general purpose language
- Python is an Object Oriented language
- [...]
- [...]
- We recommend using Python 3

March 26 2019

CSC 5741 L02 - 18

Getting Started With Python (3/3)

- Python statements can be executed directly from the interpreter
- Python scripts can be executed as shell commands

March 26 2019

CSC 5741 L02 - 19

Installation and Setup

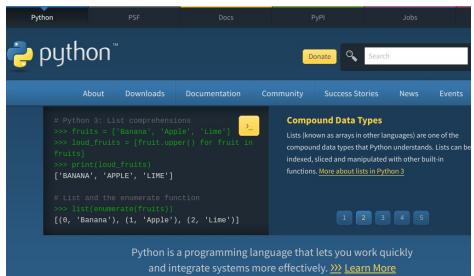
- [...]

March 26 2019

CSC 5741 L02 - 20

Installation and Setup (1/2)

- Download and install the latest version of Python 3
 - Installers also available on course Web page, in the resources directory
- Download and install the latest version of pip



A screenshot of the Python website. It shows a code editor with Python code examples. One example demonstrates list comprehensions:`>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for fruit in fruits]
["BANANA", "APPLE", "LIME"]`

The other example shows how to use the enumerate function:`>>> list(enumerate(fruits))
[(0, 'Banana'), (1, 'Apple'), (2, 'Lime')]`

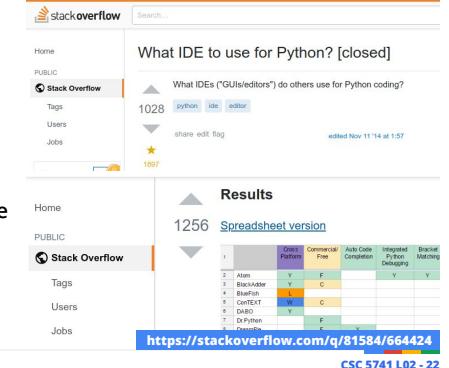
Below the code editor, there is a brief description of lists and a link to learn more.

March 26 2019

CSC 5741 L02 - 21

Installation and Setup (2/2)

- Any text editor will be sufficient for scripting.
 - Vim, Notepad [...]
- On IDEs
 - There are plenty of IDEs to choose from
 - In the recent past, I have worked with Wing 101 and Kate



A screenshot of a Stack Overflow search results page. The search query is "What IDEs ('GUIs/editors') do others use for Python coding?". The results table shows various IDEs and their features:

| ID | Name | Mac | Platform | Commercial | Auto Code Completion | Integrated Python | Bracket Matching |
|----|-----------|-----|----------|------------|----------------------|-------------------|------------------|
| 1 | PyCharm | Y | Y | F | C | Y | Y |
| 2 | Atom | Y | Y | F | C | Y | Y |
| 3 | Blender | Y | Y | F | C | Y | Y |
| 4 | Bluefish | Y | Y | F | C | Y | Y |
| 5 | CodeEdit | Y | Y | F | C | Y | Y |
| 6 | D4BO | Y | Y | F | C | Y | Y |
| 7 | Dr Python | Y | Y | F | C | Y | Y |
| 8 | Emacs | Y | Y | F | C | Y | Y |
| 9 | Geany | Y | Y | F | C | Y | Y |
| 10 | Kate | Y | Y | F | C | Y | Y |

March 26 2019

CSC 5741 L02 - 22

Basics

- No need to specify data types on variable declaration
- Indentation is important

March 26 2019

CSC 5741 L02 - 23

Identifiers (1/2)

- Python is case-sensitive, meaning uppercase and lowercase are considered as different
 - age is different from AGE
 - favourite_course is different from Favourite_Course
- Variable names, like other identifiers, follow rules
 - can use letters, numbers or underscores
 - can't use other punctuation
 - can't start with a number
 - can't use Python keywords (reserved words)
- The assignment operator in Python is the equals sign =
 - >>> age = 19

March 26 2019

CSC 5741 L02 - 24

Identifiers (2/2)

- Python keywords (reserved words) can't be used when naming identifiers
- `>>> import keyword`
- `>>> keyword.kwlist`
- `['False', 'None', 'True', 'and', 'as', 'assert', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']`

March 26 2019

CSC 5741 L02 - 25

Comments (1/2)

- Comments are useful in explaining your code, and are ignored by the Python interpreter
- Single line comments are simply indicated with a hash # character
- Everything to the right of the hash is ignored
 - `>>> course_code = "csc5741" # creates a variable course_code`

March 26 2019

CSC 5741 L02 - 26

Comments (2/2)

- Multiple line comments are specified between sets of three quotes, ''' or """

```
'''Author: Paul Chola  
Course: CSC 5741  
Lecture #02'''
```

```
"""Author: Tasha Shamane  
Course: CSC 5741  
Lecture #02"""
```

March 26 2019

CSC 5741 L02 - 27

Data Types (1/3)

- Variables don't require explicit type declaration in Python, as in other programming languages
 - `>>> x = 5`
- There are a few basic data types in Python
 - Integers int
 - Float float
 - String str
 - Boolean bool

March 26 2019

CSC 5741 L02 - 28

Data Types (2/3)

- **Integer, whole numbers**
 - `>>> i = 23`
- **Float, floating point numbers**
 - full stop indicates decimal point
 - `>>> d = 2.345`
- **String, piece of text**
 - enclosed in single ('") or double quotes ("""")
 - `>>> x = 'CSC 5741'`
 - `>>> y = "CSC 5741"`

March 26 2019

CSC 5741 L02 - 29

Data Types (3/3)

- **Boolean, true or false**
 - values True and False, start with capital letter
 - 0, "", [], (), {}, None are considered False, everything else is True
 - `>>> weekday = True`

March 26 2019

CSC 5741 L02 - 30

Functions (1/3)

- Functions are used to perform simple operations, sometimes on values
- Functions are called with round brackets ()
 - `function_name()`
- Functions can be passed certain values, which are referred to as parameters (or arguments) separated by commas
 - `function_name(parameter)`
 - `function_name(parameter1, parameter2, ...)`

March 26 2019

CSC 5741 L02 - 31

Functions (2/3)

- Python has many built-in functions, here are some:
 - `print()` function prints information to the screen
 - `input()` function gets information from the user
 - `type()` function returns data type of variable or value
 - `>>> x = 3`
 - `>>> type(x)`
 - `<class 'int'>`

March 26 2019

CSC 5741 L02 - 32

Functions (3/3)

```
def csc5741(x, y='Y', z='Z'):
    print(x + ' ' + y)
return 0

csc5741('Xxxx', 'Yyyyy')
```

- All arguments are named
- Naming useful for optional arguments
- Return is optional

March 26 2019

CSC 5741 L02 - 33

CSC 5741 L02 - 34

Data Structures

- Tuple
 - var = (1, 2, 3, 4, 5)
- List
 - var = [1, 2, 3, 4, 5]
- Dictionary
 - var = {"one":1, "two":2, "three":3, "four":4, "five":5}
- Set
 - var = {1, 2, 3, 4, 5}

March 26 2019

CSC 5741 L02 - 34

Loops

```
for i in [1,2,3]:
    print(i)

while i < 5:
    i += 1
    print(i)
```

- No curly braces or "end for"
- Structure is derived from level of indentation
- One statement per line
- No semicolons required

March 26 2019

CSC 5741 L02 - 35

CSC 5741 L02 - 36

Modules (1/2)

- Modules facilitate extensibility and reusability
- Modules are collections of functions adding functionality to Python
- Modules can be imported using import keyword
 - Once modules are imported, their functions can be accessed by using the module name
 - The help() function displays what is contained in a module

```
from math import sqrt
import math
```

March 26 2019

CSC 5741 L02 - 36

Modules (2/2)

- Single functions can be imported using the from statement
 - `>>> from math import sqrt`
 - When using the from statement functions can be accessed without the module name
 - `>>> sqrt(16)`
 - Everything from the module can be imported using an asterisk with the from statement
 - `>>> from math import *`

March 26 2019

CSC 5741 L02 - 37

Lecture Series Outline

- **Part I: Getting Started With Python**
 - **Part II: pandas, matplotlib and scikit-learn**
 - matplotlib
 - pandas
 - scikit-learn
 - **Part III: Academic Talk [Trial]**
 - **Part IV: Paper Reading [Trial]**
 - **Part V: About Next Week**

March 26 2019

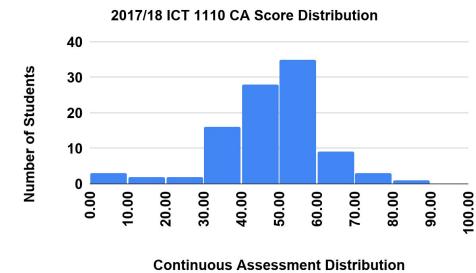
CSC 5741 L02 - 38

Matplotlib (1/7)

- The matplotlib library is best installed using pip, as with all libraries or using apt-get, if on Mac or Linux
 - pip3 install matplotlib
 - sudo apt-get install python-matplotlib
 - Test installation by importing a library module

March 26 2019

CSC 5741 L02 - 39



March 26 2019

CSC 5741 L02 - 40

Matplotlib (3/7)

- **Creating plots is a four-step process**

- 1) Import matplotlib
- 2) Draw the plot
- 3) Specify plot aesthetics
- 4) Render plot

March 26 2019

CSC 5741 L02 - 41

CSC 5741 L02 - 42

Matplotlib (4/7)

- **Creating plots is a four-step process**

- 1) Import matplotlib
import matplotlib.pyplot as plt
- 2) Draw the plot
- 3) Specify plot aesthetics
- 4) Render plot

March 26 2019

CSC 5741 L02 - 41

CSC 5741 L02 - 42

Matplotlib (5/7)

- **Creating plots is a four-step process**

- 1) Import matplotlib
import matplotlib.pyplot as plt
- 2) Draw the plot
plt.plot([...])
plt.hist([...])
- 3) Specify plot aesthetics
- 4) Render plot

March 26 2019

CSC 5741 L02 - 43

CSC 5741 L02 - 44

Matplotlib (6/7)

- **Creating plots is a four-step process**

- 1) Import matplotlib
import matplotlib.pyplot as plt
- 2) Draw the plot
plt.plot([...])
plt.hist([...])
- 3) Specify plot aesthetics
plt.xlabel(" [...] ")
plt.ylabel(" [...] ")
- 4) Render plot

- **Illustration**

- Simple plots
- Plots using pandas
dataframe

Matplotlib (7/7)

- **Creating plots is a four-step process**
 - 1) Import matplotlib
import matplotlib.pyplot as plt
 - 2) Draw the plot
plt.plot([...])
plt.hist([...])
 - 3) Specify plot aesthetics
plt.xlabel("..."); plt.ylabel("...")
plt.legend()
 - 4) Render plot
plt.show()
- **Illustration**
 - Simple plots
 - Plots using pandas dataframe

March 26 2019

CSC 5741 L02 - 45

Matplotlib—Exercise

- JCTR regularly compiles Basic Needs Basket statistics for major towns in The Republic of Zambia. Using the Lusaka April 2018 BNB dataset (dataset available on <https://goo.gl/1zSigy>):
- Draw a line plot showing the trends of the Lusaka BNB between November 2016 and April 2018
- Draw a bar plot showing BNB costs across Zambia in April 2018
- Draw a pie chart showing the cost of basic food items in Lusaka for the month of April 2018

March 26 2019

CSC 5741 L02 - 46

Lecture Series Outline

- **Part I: Getting Started With Python**
- **Part II: pandas, matplotlib and scikit-learn**
 - matplotlib
 - pandas
 - scikit-learn
- **Part III: Academic Talk [Trial]**
- **Part IV: Paper Reading [Trial]**
- **Part V: About Next Week**

March 26 2019

CSC 5741 L02 - 47

Pandas (1/9)

- **Why use pandas instead a spreadsheet for data analysis**
 - Efficiency as data scales
 - Very user-friendly
 - Dataframe similar to spreadsheet

March 26 2019

CSC 5741 L02 - 48

Pandas (2/9)

- Why use pandas instead a spreadsheet for data analysis
 - Efficiency as data scales
 - Very user-friendly
 - Dataframe similar to spreadsheet

March 26 2019

CSC 5741 L02 - 49

Pandas (3/9)

- Pandas DataFrame
 - Two dimensional labeled data structure
 - DataFrame can be viewed as a representation of a Spreadsheet worksheet

| | StudentID | Gender | Minor | LastName | ... PassedTest3 |
|----|----------------------------|--------|-------------|-----------|-----------------|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayava | ... |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... |
| 3 | 2017012993@student.unza.zm | M | Civic | Bwalya | ... |
| 4 | 2017080514@student.unza.zm | M | History | Bwalya | ... |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... |
| 6 | 2017012923@student.unza.zm | M | Civic | Chatabela | ... |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... |
| 8 | 2017012963@student.unza.zm | M | Mathematics | Chitete | ... |
| 9 | 2017080345@student.unza.zm | M | Mathematics | Chileshe | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... |
| 11 | 2017012965@student.unza.zm | F | Languages | Chisha | ... |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... |
| 13 | 2017012990@student.unza.zm | M | Mathematics | Hamamba | ... |
| 14 | 2017012912@student.unza.zm | M | Civic | Imakando | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... |

Shell: #!/bin/bash -i; cd /home/unza/lectures/lecture42/python3; ./script3

CSC 5741 L02 - 50

Pandas (4/9)

- Pandas series
 - One dimensional labeled array that can hold any data type.
 - Similar to column in Spreadsheet applications

| | StudentID | Gender | Minor | LastName | ... PassedTest3 |
|----|----------------------------|--------|-------------|-----------|-----------------|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayava | ... |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... |
| 3 | 2017012993@student.unza.zm | M | Civic | Bwalya | ... |
| 4 | 2017080514@student.unza.zm | M | History | Bwalya | ... |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... |
| 6 | 2017012923@student.unza.zm | M | Civic | Chatabela | ... |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... |
| 8 | 2017012963@student.unza.zm | M | Mathematics | Chitete | ... |
| 9 | 2017080345@student.unza.zm | M | Mathematics | Chileshe | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... |
| 11 | 2017012965@student.unza.zm | F | Languages | Chisha | ... |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... |
| 13 | 2017012990@student.unza.zm | M | Mathematics | Hamamba | ... |
| 14 | 2017012912@student.unza.zm | M | Civic | Imakando | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... |

Shell: #!/bin/bash -i; cd /home/unza/lectures/lecture42/python3; ./script3

March 26 2019

CSC 5741 L02 - 51

Pandas (5/9)

- Columns
 - Ellipse indicate more columns. Structure of data frame indicated on last line of output

| | StudentID | Gender | Minor | LastName | ... PassedTest3 |
|----|----------------------------|--------|-------------|-----------|-----------------|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayava | ... |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... |
| 3 | 2017012993@student.unza.zm | M | Civic | Bwalya | ... |
| 4 | 2017080514@student.unza.zm | M | History | Bwalya | ... |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... |
| 6 | 2017012923@student.unza.zm | M | Civic | Chatabela | ... |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... |
| 8 | 2017012963@student.unza.zm | M | Mathematics | Chitete | ... |
| 9 | 2017080345@student.unza.zm | M | Mathematics | Chileshe | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... |
| 11 | 2017012965@student.unza.zm | F | Languages | Chisha | ... |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... |
| 13 | 2017012990@student.unza.zm | M | Mathematics | Hamamba | ... |
| 14 | 2017012912@student.unza.zm | M | Civic | Imakando | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... |

Shell: #!/bin/bash -i; cd /home/unza/lectures/lecture42/python3; ./script3

CSC 5741 L02 - 52

March 26 2019

Pandas (6/9)

- **Index**

- Automatically generated, but can be changed
- Uniquely identifies rows in the DataFrame

| | StudentID | Gender | Minor | LastName | ... PassedTest3 | |
|----|----------------------------|--------|-------------|-----------|-----------------|-----|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayava | ... | NO |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 2 | 2017012202@student.unza.zm | M | Languages | Banda | ... | NO |
| 3 | 2017012911@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 4 | 2017080514@student.unza.zm | M | History | Bwalya | ... | NO |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 6 | 2017012923@student.unza.zm | M | Civic | Chabela | ... | YES |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... | NO |
| 8 | 2017012984@student.unza.zm | M | Mathematics | Chabula | ... | NO |
| 9 | 2017008343@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 11 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 12 | 2017012939@student.unza.zm | F | History | Gondwe | ... | NO |
| 13 | 2017012930@student.unza.zm | M | Mathematics | Hamamanda | ... | NO |
| 14 | 2017012320@student.unza.zm | F | Civic | Imakando | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamsanga | ... | YES |

March 26 2019

CSC 5741 L02 - 53

Pandas (7/9)

- **Data**

| | StudentID | Gender | Minor | LastName | ... PassedTest3 | |
|----|----------------------------|--------|-------------|-----------|-----------------|-----|
| 0 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 1 | 2017012911@student.unza.zm | M | Languages | Banda | ... | NO |
| 2 | 2017012983@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 3 | 2017080514@student.unza.zm | M | History | Bwalya | ... | NO |
| 4 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 5 | 2017012923@student.unza.zm | M | Civic | Chabela | ... | YES |
| 6 | 2017012984@student.unza.zm | M | History | Chakulya | ... | NO |
| 7 | 2017012985@student.unza.zm | M | Mathematics | Chabula | ... | NO |
| 8 | 2017008343@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 9 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 10 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 11 | 2017012965@student.unza.zm | F | History | Gondwe | ... | NO |
| 12 | 2017012939@student.unza.zm | M | Mathematics | Hamamanda | ... | NO |
| 13 | 2017012930@student.unza.zm | F | Civic | Imakando | ... | NO |
| 14 | 2017012320@student.unza.zm | F | Civic | Indakodo | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamsanga | ... | NO |

March 26 2019

CSC 5741 L02 - 54

Pandas (8/9)

- **Some common operations**
 - Reading data files
 - `df.read_csv([...])`
 - `df.read_html([...])`
 - `df.read_json([...])`
 - `df.read_*`
 - Inspecting dataframes
 - `df.head([...])`
 - `df.tail([...])`
 - `df.columns`
 - `df['...']`

March 26 2019

CSC 5741 L02 - 55

Pandas (9/9)

- **Some common operations**
 - Converting to different file formats
 - `df.to_csv([...])`
 - `df.to_excel([...])`
 - `df.to_sql([...])`
 - `df.to_*`
 - Renaming columns
 - `df.rename(columns={...})`
 - Aggregating data
 - `df.groupby(['...']).mean()`
 - `df.groupby(['...']).max()`

March 26 2019

CSC 5741 L02 - 56

Pandas—Exercise

- Using pandas and the 2018/19 ICT 1110 assessment scores dataset (dataset available on <https://goo.gl/wC1H7Q>):
 - Print out the details (using a List) of the student who got the highest CA score
 - Print out the average CA scores for each of the different Minors
 - Print out the average CA scores for each gender
 - Export a summary table, to HTML, of mean CA scores by student Minors
 - Export, to CSV, a dataset consisting of StudentID, Minor and Total CA

March 26 2019

CSC 5741 L02 - 57

CSC 5741 L02 - 58

Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
 - matplotlib
 - pandas
 - scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week

March 26 2019

CSC 5741 L02 - 58

Scikit-learn

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
 - matplotlib
 - pandas
 - scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week

March 26 2019

CSC 5741 L02 - 59

CSC 5741 L02 - 60

Scikit-learn

- Scikit-learn
 - Ensure that the module is installed by using the import statement

```
lightonphiri@lightonphiri-Lenovo-ideapad-320-15IKB:~$ pip3 install sklearn
Collecting sklearn
  Downloading https://files.pythonhosted.org/packages/1e/7a/dbb3be0ce9bd5c8b7e3d
sklearn-0.0.tar.gz
Collecting scikit-learn (from sklearn)
  Downloading https://files.pythonhosted.org/packages/5c/82/c0de5839d613b82bdd0
scikit_learn-0.20.3-cp36-cp36-manylinux1_x86_64.whl (5.4MB)
    0% |████████████████████████████████| 20KB 55KB/s eta 0:01:38
```

```
lightonphiri@lightonphiri-Lenovo-ideapad-320-15IKB:~$ python3
Python 3.6.7 (default, Oct 22 2018, 11:32:17)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import sklearn
>>> dir(sklearn)
['__SKLEARN_SETUP__', '__all__', '__builtins__', '__cached__', '__check_build__', '__doc__', '__file__', '__loader__', '__name__', '__package__', '__path__', '__spec__', '__version__', '__config__', 'base', 'clone', 'externals', 'get_config', 'logger', 'logging', 're', 'set_config', 'setup_module', 'show_versions']
>>> __
```

Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
 - Towards Increased Online Visibility of Research in Zambia
- Part IV: Paper Reading [Trial]
- Part V: About Next Week

March 26 2019

CSC 5741 L02 - 61

Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
 - L. Phiri (2018) "Research Visibility in the Global South: Towards Increased Online Visibility of Scholarly Research Output in Zambia"
- Part V: About Next Week

March 26 2019

CSC 5741 L02 - 62

Academic Talk [Trial]

Research Visibility in the Global South: Towards Increased Online Visibility of Research in Zambia

Lighton Phiri <lighton.phiri@unza.zm>

Department of Library and Information Science
University of Zambia

March 26 2019

CSC 5741 L02 - 63

Paper Reading Session [Trial]

- [1] L. Phiri (2018) "Research Visibility in the Global South: Towards Increased Online Visibility of Scholarly Research Output in Zambia"
http://lis.unza.zm/~lightonphiri/papers/paper-icict18-online_visibility.pdf

March 26 2019

CSC 5741 L02 - 64

Bibliography

- [1] Python for Beginners | Python.org
<https://www.python.org/about/gettingstarted>
- [2] Pyplot tutorial - Matplotlib 3.0.3 documentation
<https://matplotlib.org/tutorials/introductory/pyplot.html>
- [3] 10 Minutes to pandas - pandas 0.22.0 documentation
<https://pandas.pydata.org/pandas-docs/version/0.22/10min.html>

March 26 2019

CSC 5741 L02 - 65



CSC 5741

Lecture 2: Python for Data Mining and Machine Learning



Lighton Phiri <lighton.phiri@unza.zm>

Department of Library and Information Science
University of Zambia

