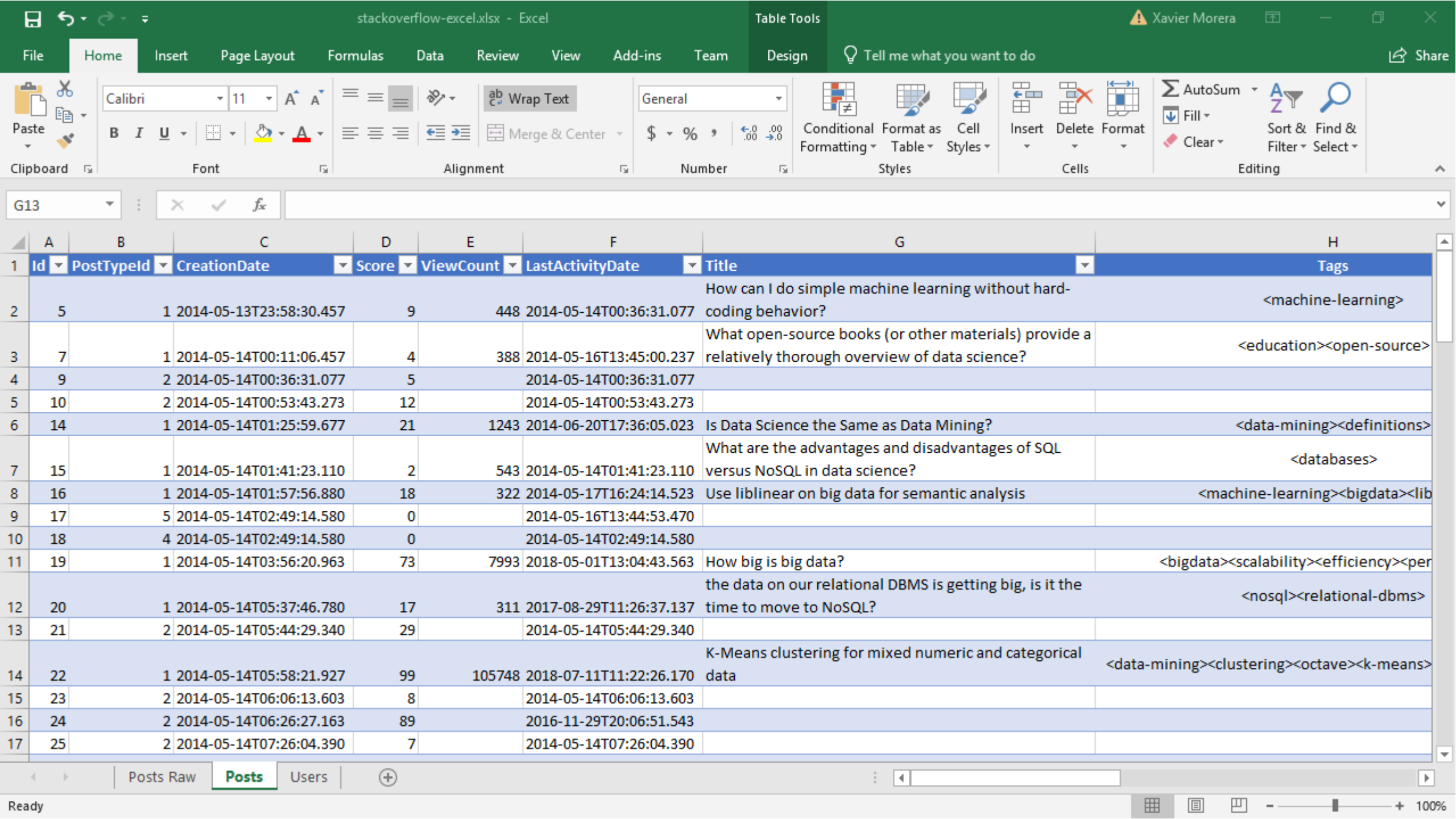
Importing Data into Python from Excel Files

Xavier Morera
BIG DATA INC.

@xmorera www.xaviermorera.com



	A	B	C	D	E	F	G	H
1	Id	PostTypeId	CreationDate	Score	ViewCount	LastActivityDate	Title	Tags
2	5	1	2014-05-13T23:58:30.457	9	448	2014-05-14T00:36:31.077	How can I do simple machine learning without hard-coding behavior?	<machine-learning>
3	7	1	2014-05-14T00:11:06.457	4	388	2014-05-16T13:45:00.237	What open-source books (or other materials) provide a relatively thorough overview of data science?	<education><open-source>
4	9	2	2014-05-14T00:36:31.077	5		2014-05-14T00:36:31.077		
5	10	2	2014-05-14T00:53:43.273	12		2014-05-14T00:53:43.273		
6	14	1	2014-05-14T01:25:59.677	21	1243	2014-06-20T17:36:05.023	Is Data Science the Same as Data Mining?	<data-mining><definitions>
7	15	1	2014-05-14T01:41:23.110	2	543	2014-05-14T01:41:23.110	What are the advantages and disadvantages of SQL versus NoSQL in data science?	<databases>
8	16	1	2014-05-14T01:57:56.880	18	322	2014-05-17T16:24:14.523	Use liblinear on big data for semantic analysis	<machine-learning><bigdata><lib
9	17	5	2014-05-14T02:49:14.580	0		2014-05-16T13:44:53.470		
10	18	4	2014-05-14T02:49:14.580	0		2014-05-14T02:49:14.580		
11	19	1	2014-05-14T03:56:20.963	73	7993	2018-05-01T13:04:43.563	How big is big data?	<bigdata><scalability><efficiency><per
12	20	1	2014-05-14T05:37:46.780	17	311	2017-08-29T11:26:37.137	the data on our relational DBMS is getting big, is it the time to move to NoSQL?	<nosql><relational-dbms>
13	21	2	2014-05-14T05:44:29.340	29		2014-05-14T05:44:29.340		
14	22	1	2014-05-14T05:58:21.927	99	105748	2018-07-11T11:22:26.170	K-Means clustering for mixed numeric and categorical data	<data-mining><clustering><octave><k-means>
15	23	2	2014-05-14T06:06:13.603	8		2014-05-14T06:06:13.603		
16	24	2	2014-05-14T06:26:27.163	89		2016-11-29T20:06:51.543		
17	25	2	2014-05-14T07:26:04.390	7		2014-05-14T07:26:04.390		

```
import pandas as pd
excel_file = pd.ExcelFile('stackoverflow.xlsx')
type(excel_file)
excel_file.sheet_names
excel_df = excel_file.parse()
type(excel_df)
excel_df.head()
excel_df
```

Importing Excel Files Using Pandas

Start by importing pandas

- Requires xlrd

Load into ExcelFile

- Use parse

```
posts_excel = pd.read_excel('stackoverflow-one.xlsx')
type(posts_excel)
dir(posts_excel)
posts_excel.columns
posts_excel.head()
pd.read_excel('stackoverflow-one.xlsx', usecols=[0, 3]).columns
pd.read_excel('stackoverflow-one.xlsx', usecols='A:C').columns
pd.read_excel('stackoverflow-one.xlsx', usecols='A,C').columns
```

Importing Excel Files Using Pandas

Load using read_excel

- DataFrame
- Similar process as read_csv and read_table
- Also dict of DataFrame

```
excel_file.sheet_names
posts_dict = pd.read_excel('stackoverflow.xlsx',sheet_name=None)
type(posts_dict)
posts_dict.keys()
posts_dict['Posts'].head()
```

Importing Excel Files Using Pandas

Use sheet_name for selecting sheet

Many more parameters available

- i.e. usecols, dtype, skiprows, nrows, header...

```
posts_dict['Users'].head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Users').head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', usecols=range(1,9)).head()
pd.read_excel('stackoverflow.xlsx',sheet_name=2).head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', usecols=range(1,9)).head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', usecols=range(1,9),skiprows=4).head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', usecols=range(1,9),nrows=2).head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', usecols=range(1,9)).dtypes
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', usecols=range(1,9), dtype={'PostTypeId':
str}).dtypes
pd.read_excel('stackoverflow.xlsx',sheet_name='Users', converters={'Id': lambda x: x + 1000}).head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Posts', usecols=[0,7,8]).head()
pd.read_excel('stackoverflow.xlsx',sheet_name='Posts', usecols=[0,7,8], keep_default_na=False).head()
```

pandas.read_excel

pandas.**read_excel**(*io*, *sheet_name=0*, *header=0*, *names=None*, *index_col=None*, *usecols=None*, *squeeze=False*, *dtype=None*, *engine=None*, *converters=None*, *true_values=None*, *false_values=None*, *skiprows=None*, *nrows=None*, *na_values=None*, *parse_dates=False*, *date_parser=None*, *thousands=None*, *comment=None*, *skipfooter=0*, *convert_float=True*, ***kwds*) [source]

Read an Excel table into a pandas DataFrame

io : *string, path object (pathlib.Path or py._path.local.LocalPath),*
file-like object, pandas ExcelFile, or xlrd workbook. The string could be a URL. Valid URL schemes include http, ftp, s3, and file. For file URLs, a host is expected. For instance, a local file could be file://localhost/path/to/workbook.xlsx

sheet_name : *string, int, mixed list of strings/ints, or None, default 0*
Strings are used for sheet names, Integers are used in zero-indexed sheet positions. Lists of strings/integers are used to request multiple sheets.
Specify None to get all sheets.
str|int -> DataFrame is returned. list|None -> Dict of DataFrames is returned, with keys representing sheets.
Available Cases
- Defaults to 0 -> 1st sheet as a DataFrame
- 1 -> 2nd sheet as a DataFrame
- "Sheet1" -> 1st sheet as a DataFrame
- [0,1,"Sheet5"] -> 1st, 2nd & 5th sheet as a dictionary of DataFrames
- None -> All sheets as a dictionary of DataFrames

sheetname : *string, int, mixed list of strings/ints, or None, default 0*

Deprecated since version 0.21.0: Use *sheet_name* instead

header : *int, list of ints, default 0*
Row (0-indexed) to use for the column labels of the parsed DataFrame. If a list of integers is passed those row positions will be combined into a `MultiIndex`. Use None