**pandas.read\_csv**参数整理

读取CSV（逗号分割）文件到DataFrame

也支持文件的部分导入和选择迭代

更多帮助参见：[http://pandas.pydata.org/pandas-docs/stable/io.html](http://pandas.pydata.org/pandas-docs/stable/io.html" \t "http://blog.csdn.net/sinat_36709248/article/details/_blank)

参数：

**filepath\_or\_buffer** : str，pathlib。str, pathlib.Path, py.\_path.local.LocalPath or any object with a read() method (such as a file handle or StringIO)

可以是URL，可用URL类型包括：http, ftp, s3和文件。对于多文件正在准备中

本地文件读取实例：://localhost/path/to/table.csv

**sep**: str, default ‘,’

指定分隔符。如果不指定参数，则会尝试使用逗号分隔。分隔符长于一个字符并且不是‘\s+’,将使用python的语法分析器。并且忽略数据中的逗号。正则表达式例子：'\r\t'

**delimiter**: str, default None

定界符，备选分隔符（如果指定该参数，则sep参数失效）

**delim\_whitespace** : boolean, default False.

指定空格(例如’ ‘或者’ ‘)是否作为分隔符使用，等效于设定sep='\s+'。如果这个参数设定为Ture那么delimiter 参数失效。

**header** : int or list of ints, default ‘infer’

指定行数用来作为列名，数据开始行数。如果文件中没有列名，则默认为0，否则设置为None。如果明确设定header=0 就会替换掉原来存在列名。header参数可以是一个list例如：[0,1,3]，这个list表示将文件中的这些行作为列标题（意味着每一列有多个标题），介于中间的行将被忽略掉（例如本例中的2；本例中的数据1,2,4行将被作为多级标题出现，第3行数据将被丢弃，dataframe的数据从第5行开始。）。

注意：如果skip\_blank\_lines=True 那么header参数忽略注释行和空行，所以header=0表示第一行数据而不是文件的第一行。

**names**: array-like, default None

用于结果的列名列表，如果数据文件中没有列标题行，就需要执行header=None。默认列表中不能出现重复，除非设定参数mangle\_dupe\_cols=True。

**index\_col** : int or sequence or False, default None

用作行索引的列编号或者列名，如果给定一个序列则有多个行索引。

如果文件不规则，行尾有分隔符，则可以设定index\_col=False 来是的pandas不适用第一列作为行索引。

**usecols** : array-like, default None

返回一个数据子集，该列表中的值必须可以对应到文件中的位置（数字可以对应到指定的列）或者是字符传为文件中的列名。例如：usecols有效参数可能是 [0,1,2]或者是 [‘foo’, ‘bar’, ‘baz’]。使用这个参数可以加快加载速度并降低内存消耗。

**as\_recarray** : boolean, default False

不赞成使用：该参数会在未来版本移除。请使用pd.read\_csv(...).to\_records()替代。

返回一个Numpy的recarray来替代DataFrame。如果该参数设定为True。将会优先squeeze参数使用。并且行索引将不再可用，索引列也将被忽略。

**squeeze**: boolean, default False

如果文件值包含一列，则返回一个Series

**prefix**: str, default None

在没有列标题时，给列添加前缀。例如：添加‘X’ 成为 X0, X1, ...

**mangle\_dupe\_cols**: boolean, default True

重复的列，将‘X’...’X’表示为‘X.0’...’X.N’。如果设定为false则会将所有重名列覆盖。

**dtype** : Type name or dict of column -> type, default None

每列数据的数据类型。例如 {‘a’: np.float64, ‘b’: np.int32}

**engine**: {‘c’, ‘python’}, optional

Parser engine to use. The C engine is faster while the python engine is currently more feature-complete.

使用的分析引擎。可以选择C或者是python。C引擎快但是Python引擎功能更加完备。

**converters** : dict, default None

列转换函数的字典。key可以是列名或者列的序号。

**true\_values** : list, default None Values to consider as True

**false\_values** : list, default None Values to consider as False

**skipinitialspace**: boolean, default False

忽略分隔符后的空白（默认为False，即不忽略）.

**skiprows** : list-like or integer, default None

需要忽略的行数（从文件开始处算起），或需要跳过的行号列表（从0开始）。

**skipfooter** : int, default 0

从文件尾部开始忽略。 (c引擎不支持)

**skip\_footer** : int, default 0

不推荐使用：建议使用skipfooter ，功能一样。

**nrows** : int, default None

需要读取的行数（从文件头开始算起）。

**na\_values** : scalar, str, list-like, or dict, default None

一组用于替换NA/NaN的值。如果传参，需要制定特定列的空值。默认为‘1.#IND’, ‘1.#QNAN’, ‘N/A’, ‘NA’, ‘NULL’, ‘NaN’, ‘nan’`.

**keep\_default\_na**: bool, default True

如果指定na\_values参数，并且keep\_default\_na=False，那么默认的NaN将被覆盖，否则添加。

**na\_filter**: boolean, default True

是否检查丢失值（空字符串或者是空值）。对于大文件来说数据集中没有空值，设定na\_filter=False可以提升读取速度。

**verbose** : boolean, default False

是否打印各种解析器的输出信息，例如：“非数值列中缺失值的数量”等。

**skip\_blank\_lines** : boolean, default True

如果为True，则跳过空行；否则记为NaN。

**parse\_dates**: boolean or list of ints or names or list of lists or dict, default False

boolean. True -> 解析索引

list of ints or names. e.g. If [1, 2, 3] -> 解析1,2,3列的值作为独立的日期列

list of lists. e.g. If [[1, 3]] -> 合并1,3列作为一个日期列使用

dict, e.g. {‘foo’ : [1, 3]} -> 将1,3列合并，并给合并后的列起名为"foo"

**infer\_datetime\_format** : boolean, default False

如果设定为True并且parse\_dates 可用，那么pandas将尝试转换为日期类型，如果可以转换，转换方法并解析。在某些情况下会快5~10倍。

**keep\_date\_col**: boolean, default False

如果连接多列解析日期，则保持参与连接的列。默认为False。

**date\_parser** : function, default None

用于解析日期的函数，默认使用dateutil.parser.parser来做转换。Pandas尝试使用三种不同的方式解析，如果遇到问题则使用下一种方式。

1.使用一个或者多个arrays（由parse\_dates指定）作为参数；

2.连接指定多列字符串作为一个列作为参数；

3.每行调用一次date\_parser函数来解析一个或者多个字符串（由parse\_dates指定）作为参数。

**dayfirst**: boolean, default False DD/MM格式的日期类型

**iterator**: boolean, default False

返回一个Text，File，Reader 对象，以便逐块处理文件。

**chunksize** : int, default None

文件块的大小， [See IO Tools docs for more information](http://pandas.pydata.org/pandas-docs/stable/io.html" \l "io-chunking" \t "http://blog.csdn.net/sinat_36709248/article/details/_blank)on iterator and chunksize.

**compression** : {‘infer’, ‘gzip’, ‘bz2’, ‘zip’, ‘xz’, None}, default ‘infer’

直接使用磁盘上的压缩文件。如果使用infer参数，则使用 gzip, bz2, zip或者解压文件名中以‘.gz’, ‘.bz2’, ‘.zip’, or ‘xz’这些为后缀的文件，否则不解压。如果使用zip，那么ZIP包中国必须只包含一个文件。设置为None则不解压。

新版本0.18.1版本支持zip和xz解压

**thousands** : str, default None

千分位分割符，如“，”或者“."

**decimal** : str, default ‘.’

字符中的小数点 (例如：欧洲数据使用’，‘).

**float\_precision** : string, default None 指定

**lineterminator**: str (length 1), default None

行分割符，只在C解析器下使用。

**quotechar**: str (length 1), optional

引号，用作标识开始和解释的字符，引号内的分割符将被忽略。

**quoting** : int or csv.QUOTE\_\* instance, default 0

控制csv中的引号常量。可选 QUOTE\_MINIMAL (0), QUOTE\_ALL (1), QUOTE\_NONNUMERIC (2) or QUOTE\_NONE (3)

**doublequote** : boolean, default True

双引号，当单引号已经被定义，并且quoting 参数不是QUOTE\_NONE的时候，使用双引号表示引号内的元素作为一个元素使用。

**escapechar** : str (length 1), default None

当quoting 为QUOTE\_NONE时，指定一个字符使的不受分隔符限值。

**comment** : str, default None

标识着多余的行不被解析。如果该字符出现在行首，这一行将被全部忽略。这个参数只能是一个字符，空行（就像skip\_blank\_lines=True）注释行被header和skiprows忽略一样。例如如果指定comment='#' 解析‘#empty\na,b,c\n1,2,3’ 以header=0 那么返回结果将是以’a,b,c'作为header。

**encoding** : str, default None

指定字符集类型，通常指定为'utf-8'.

**dialect** : str or csv.Dialect instance, default None

如果没有指定特定的语言，如果sep大于一个字符则忽略。具体查看csv.Dialect 文档

**tupleize\_cols** : boolean, default False

**error\_bad\_lines** : boolean, default True

如果一行包含太多的列，那么默认不会返回DataFrame ，如果设置成false，那么会将改行剔除（只能在C解析器下使用）。

**warn\_bad\_lines** : boolean, default True

如果error\_bad\_lines =False，并且warn\_bad\_lines =True 那么所有的“bad lines”将会被输出（只能在C解析器下使用）。

**low\_memory**: boolean, default True

分块加载到内存，再低内存消耗中解析。但是可能出现类型混淆。确保类型不被混淆需要设置为False。或者使用dtype 参数指定类型。注意使用chunksize 或者iterator 参数分块读入会将整个文件读入到一个Dataframe，而忽略类型（只能在C解析器中有效）

**memory\_map** : boolean, default False

如果使用的文件在内存内，那么直接map文件使用。使用这种方式可以避免文件再次进行IO操作。

**buffer\_lines**: int, default None

不推荐使用，这个参数将会在未来版本移除，因为他的值在解析器中不推荐使用

**compact\_ints** : boolean, default False

不推荐使用，这个参数将会在未来版本移除

如果设置compact\_ints=True ，那么任何有整数类型构成的列将被按照最小的整数类型存储，是否有符号将取决于use\_unsigned 参数

**use\_unsigned** : boolean, default False

不推荐使用：这个参数将会在未来版本移除

如果整数列被压缩(i.e. compact\_ints=True)，指定被压缩的列是有符号还是无符号的。