

[Notebook](#)[Search kaggle](#)[Data](#)[/41](#)[Output](#)[Comments \(677\)](#)[Log](#)[Versions \(740\)](#)[Forks \(7,360\)](#)[Fork Notebook](#)

DATAI

Data ScienceTutorial for Beginners

last run 2 months ago · IPython Notebook HTML · 148,342 views
using data from [Pokemon- Weedle's Cave](#) · Public

2087
voters



Tags

[beginner](#) [data visualization](#) [eda](#) [tutorial](#)

Notebook

DATA SCIENTIST

In this tutorial, I only explain you what you need to be a data scientist neither more nor less.

Data scientist need to have these skills:

1. Basic Tools: Like python, R or SQL. You do not need to know everything. What you only need is to learn how to use **python**
2. Basic Statistics: Like mean, median or standard deviation. If you know basic statistics, you can use **python** easily.
3. Data Munging: Working with messy and difficult data. Like inconsistent date and string formatting. As you guess, **python** helps us.
4. Data Visualization: Title is actually explanatory. We will visualize the data with **python** like matplotlib and seaborn libraries.
5. Machine Learning: You do not need to understand math behind the machine learning technique. You only need is understanding basics of machine learning and learning how to implement it while using **python**.

As a summary we will learn python to be data scientist !!!

Content:

1. Introduction to Python:
 - A. Matplotlib
 - B. Dictionaries
 - C. Pandas
 - D. Logic, control flow and filtering
 - E. Loop data structures
2. Python Data Science Toolbox:
 - A. User defined function
 - B. Scope
 - C. Nested function
 - D. Default and flexible arguments
 - E. Lambda function
 - F. Anonymous function
 - G. Iterators
 - H. List comprehension
3. Cleaning Data
 - A. Diagnose data for cleaning
 - B. Exploratory data analysis
 - C. Visual exploratory data analysis
 - D. Tidy data
 - E. Pivoting data
 - F. Concatenating data
 - G. Data types
 - H. Missing data and testing with assert
4. Pandas Foundation
 - A. Review of pandas
 - B. Building data frames from scratch
 - C. Visual exploratory data analysis

- D. Statistical exploratory data analysis
 - E. Indexing pandas time series
 - F. Resampling pandas time series
5. Manipulating Data Frames with Pandas
- A. Indexing data frames
 - B. Slicing data frames
 - C. Filtering data frames
 - D. Transforming data frames
 - E. Index objects and labeled data
 - F. Hierarchical indexing
 - G. Pivoting data frames
 - H. Stacking and unstacking data frames
 - I. Melting data frames
 - J. Categoricals and groupby
6. Data Visualization
- A. Seaborn: [\(https://www.kaggle.com/kanncaa1/seaborn-for-beginners\)](https://www.kaggle.com/kanncaa1/seaborn-for-beginners)
 - B. Bokeh 1: [\(https://www.kaggle.com/kanncaa1/interactive-bokeh-tutorial-part-1\)](https://www.kaggle.com/kanncaa1/interactive-bokeh-tutorial-part-1)
 - C. Rare Visualization: [\(https://www.kaggle.com/kanncaa1/rare-visualization-tools\)](https://www.kaggle.com/kanncaa1/rare-visualization-tools)
 - D. Plotly: [\(https://www.kaggle.com/kanncaa1/plotly-tutorial-for-beginners\)](https://www.kaggle.com/kanncaa1/plotly-tutorial-for-beginners)
7. Machine Learning
- A. [\(https://www.kaggle.com/kanncaa1/machine-learning-tutorial-for-beginners/\)](https://www.kaggle.com/kanncaa1/machine-learning-tutorial-for-beginners/)
8. Deep Learning
- A. [\(https://www.kaggle.com/kanncaa1/deep-learning-tutorial-for-beginners\)](https://www.kaggle.com/kanncaa1/deep-learning-tutorial-for-beginners)
9. Time Series Prediction
- A. [\(https://www.kaggle.com/kanncaa1/time-series-prediction-tutorial-with-eda\)](https://www.kaggle.com/kanncaa1/time-series-prediction-tutorial-with-eda)
10. Statistic
- A. [\(https://www.kaggle.com/kanncaa1/basic-statistic-tutorial-for-beginners\)](https://www.kaggle.com/kanncaa1/basic-statistic-tutorial-for-beginners)
11. Deep Learning with Pytorch
- A. Artificial Neural Network: [\(https://www.kaggle.com/kanncaa1/pytorch-tutorial-for-deep-learning-lovers\)](https://www.kaggle.com/kanncaa1/pytorch-tutorial-for-deep-learning-lovers)
 - B. Convolutional Neural Network: [\(https://www.kaggle.com/kanncaa1/pytorch-tutorial-for-deep-learning-lovers\)](https://www.kaggle.com/kanncaa1/pytorch-tutorial-for-deep-learning-lovers)
 - C. Recurrent Neural Network: [\(https://www.kaggle.com/kanncaa1/recurrent-neural-network-with-pytorch\)](https://www.kaggle.com/kanncaa1/recurrent-neural-network-with-pytorch)

In [1]:

```
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python docker image: https://github.com/kaggle/docker-python
# You should generally not need to make changes here, unless you are
# particularly熟悉于Python的环境，建议不要修改此部分。
```

```
com/kaggle/docker-python
# For example, here's several helpful packages to load in

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
import seaborn as sns # visualization tool

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Enter) will list the files in the input directory

from subprocess import check_output
print(check_output(["ls", "../input"]).decode("utf8"))

# Any results you write to the current directory are saved as output.
```

combats.csv
pokemon.csv
tests.csv

In [2]:

```
data = pd.read_csv('../input/pokemon.csv')
```

In [3]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800 entries, 0 to 799
Data columns (total 12 columns):
 #           800 non-null int64
 Name         799 non-null object
 Type 1      800 non-null object
 Type 2      414 non-null object
 HP           800 non-null int64
 Attack       800 non-null int64
 Defense      800 non-null int64
 Sp. Atk      800 non-null int64
 Sp. Def      800 non-null int64
 Speed         800 non-null int64
 Generation    800 non-null int64
 Legendary     800 non-null bool
 dtypes: bool(1), int64(8), object(3)
memory usage: 69.6+ KB
```

In [4]:

```
data.corr()
```

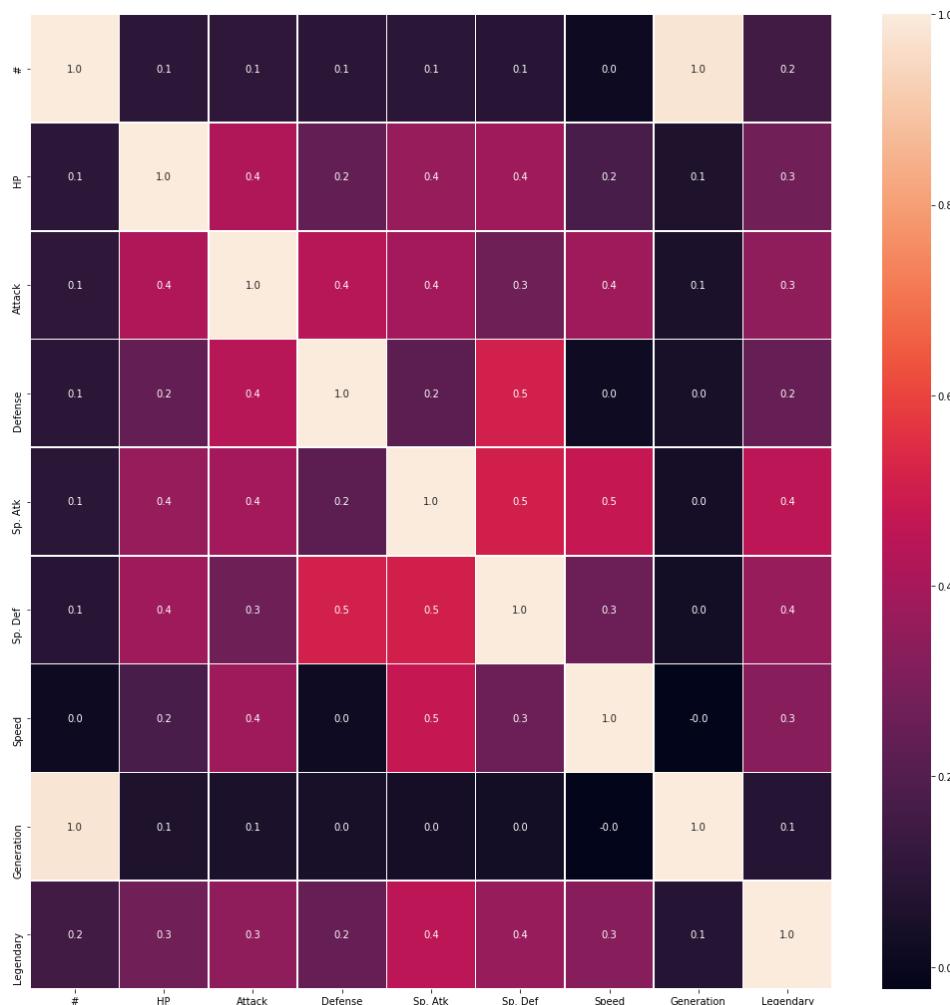
Out[4]:

	#	HP	Attack	Defense	Sp Atk	Sp Def	Speed
--	---	----	--------	---------	--------	--------	-------

	#	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legendary
#	1.000000	0.097712	0.102664	0.094691	0.089199	0.085596	0.012181		
HP	0.097712	1.000000	0.422386	0.239622	0.362380	0.378718	0.175952		
Attack	0.102664	0.422386	1.000000	0.438687	0.396362	0.263990	0.381240		
Defense	0.094691	0.239622	0.438687	1.000000	0.223549	0.510747	0.015227		
Sp. Atk	0.089199	0.362380	0.396362	0.223549	1.000000	0.506121	0.473018		
Sp. Def	0.085596	0.378718	0.263990	0.510747	0.506121	1.000000	0.259133		
Speed	0.012181	0.175952	0.381240	0.015227	0.473018	0.259133	1.000000		
Generation	0.983428	0.058683	0.051451	0.042419	0.036437	0.028486	-0.023121		
Legendary	0.154336	0.273620	0.345408	0.246377	0.448907	0.363937	0.326715		

In [5]:

```
#correlation map
f,ax = plt.subplots(figsize=(18, 18))
sns.heatmap(data.corr(), annot=True, linewidths=.5, fmt= '.1f', ax=ax)
plt.show()
```



In [6]:

```
data.head(10)
```

Out[6]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gen
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1
3	4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1
4	5	Charmander	Fire	NaN	39	52	43	60	50	65	1
5	6	Charmeleon	Fire	NaN	58	64	58	80	65	80	1
6	7	Charizard	Fire	Flying	78	84	78	109	85	100	1
7	8	Mega Charizard X	Fire	Dragon	78	130	111	130	85	100	1
8	9	Mega Charizard Y	Fire	Flying	78	104	78	159	115	100	1
9	10	Squirtle	Water	NaN	44	48	65	50	64	43	1

In [7]:

data.columns

Out[7]:

```
Index(['#', 'Name', 'Type 1', 'Type 2', 'HP', 'Attack', 'Defense',
       'Sp. Atk',
       'Sp. Def', 'Speed', 'Generation', 'Legendary'],
      dtype='object')
```

1. INTRODUCTION TO PYTHON

MATPLOTLIB

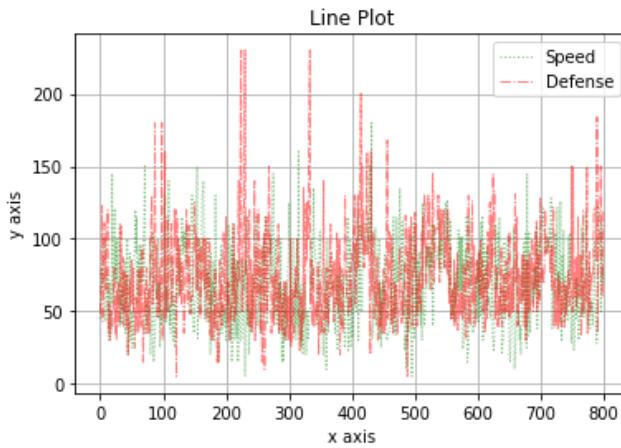
Matplot is a python library that help us to plot data. The easiest and basic plots are line, scatter and histogram plots.

- Line plot is better when x axis is time.
- Scatter is better when there is correlation between two variables
- Histogram is better when we need to see distribution of numerical data.
- Customization: Colors,labels,thickness of line, title, opacity, grid, figsize, ticks of axis and linestyle

In [8]:

```
# Line Plot
# color = color, label = label, linewidth = width of line, alpha =
# opacity, grid = grid, linestyle = style of line
data.Speed.plot(kind = 'line', color = 'g',label = 'Speed',linewi
```

```
dth=1, alpha = 0.5, grid = True, linestyle = ':')
data.Defense.plot(color = 'r', label = 'Defense', linewidth=1, alpha = 0.5, grid = True, linestyle = '-.')
plt.legend(loc='upper right')      # legend = puts label into plot
plt.xlabel('x axis')              # label = name of label
plt.ylabel('y axis')
plt.title('Line Plot')           # title = title of plot
plt.show()
```

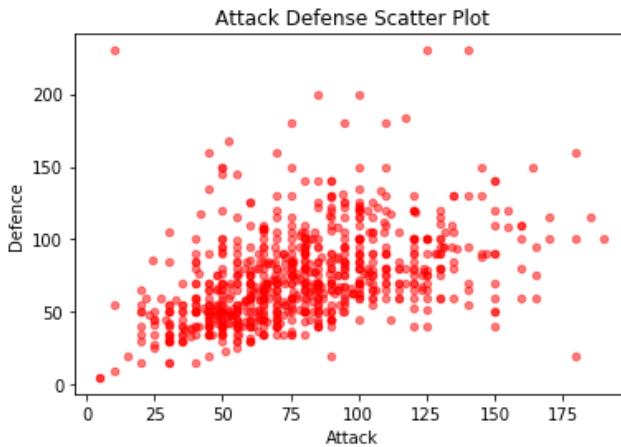


In [9]:

```
# Scatter Plot
# x = attack, y = defense
data.plot(kind='scatter', x='Attack', y='Defense', alpha = 0.5, color = 'red')
plt.xlabel('Attack')                  # label = name of label
plt.ylabel('Defence')                # label = name of label
plt.title('Attack Defense Scatter Plot') # title = title of plot
```

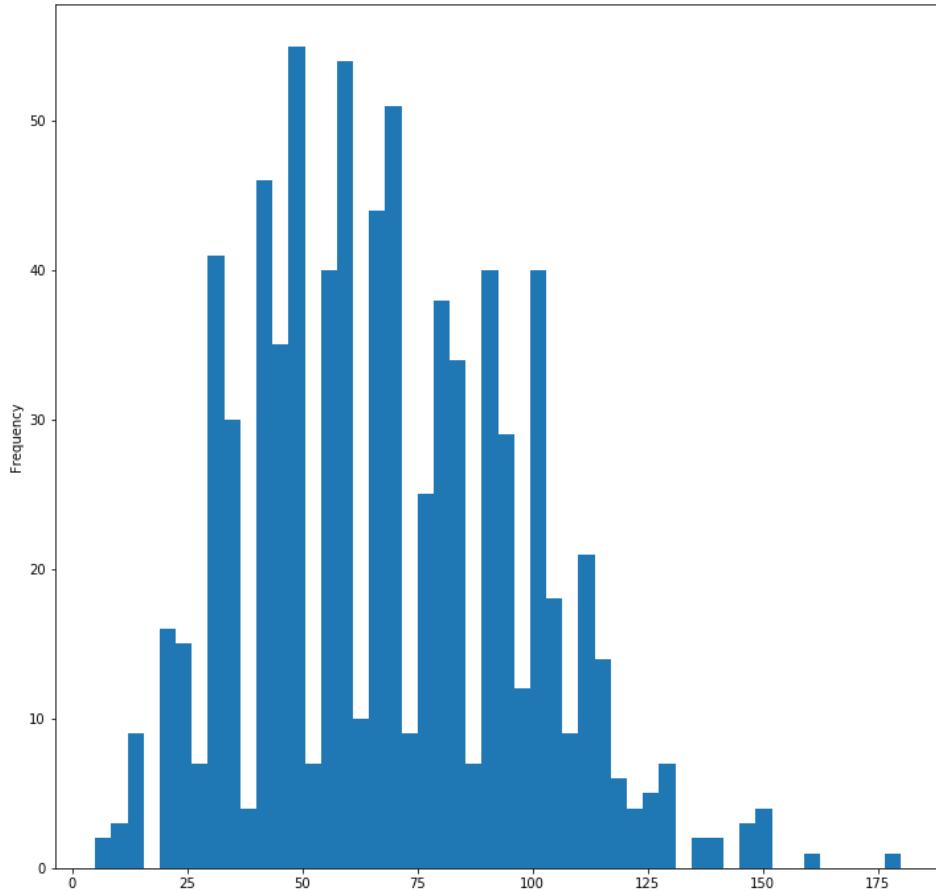
Out[9]:

Text(0.5, 1, 'Attack Defense Scatter Plot')



In [10]:

```
# Histogram
# bins = number of bar in figure
data.Speed.plot(kind = 'hist', bins = 50, figsize = (12,12))
plt.show()
```



In [11]:

```
# clf() = cleans it up again you can start a fresh
data.Speed.plot(kind = 'hist', bins = 50)
plt.clf()
# We cannot see plot due to clf()
```

<matplotlib.figure.Figure at 0x7ff3c40e3048>

DICTIONARY

Why we need dictionary?

- It has 'key' and 'value'
- Faster than lists

What is key and value. Example:

- dictionary = {'spain' : 'madrid'}
- Key is spain.
- Values is madrid.

It's that easy.

Lets practice some other properties like keys(), values(), update, add, check, remove key, remove all entries and remove dicrionary.

In [12]:

```
#create dictionary and look its keys and values
dictionary = {'spain' : 'madrid', 'usa' : 'vegas'}
print(dictionary.keys())
print(dictionary.values())
```

```
dict_keys(['spain', 'usa'])
dict_values(['madrid', 'vegas'])
```

In [13]:

```
# Keys have to be immutable objects like string, boolean, float, integer or tuples
# List is not immutable
# Keys are unique
dictionary['spain'] = "barcelona"      # update existing entry
print(dictionary)
dictionary['france'] = "paris"         # Add new entry
print(dictionary)
del dictionary['spain']                # remove entry with key 'spain'
print(dictionary)
print('france' in dictionary)          # check include or not
dictionary.clear()                     # remove all entries in dict
print(dictionary)
```

```
{'spain': 'barcelona', 'usa': 'vegas'}
{'spain': 'barcelona', 'usa': 'vegas', 'france': 'paris'}
{'usa': 'vegas', 'france': 'paris'}
True
{}
```

In [14]:

```
# In order to run all code you need to take comment this line
# del dictionary      # delete entire dictionary
print(dictionary)      # it gives error because dictionary is deleted
```

```
{}
```

PANDAS

What we need to know about pandas?

- CSV: comma - separated values

In [15]:

```
data = pd.read_csv('../input/pokemon.csv')
```

```
In [10]:  
series = data['Defense']          # data['Defense'] = series  
print(type(series))  
data_frame = data[['Defense']]   # data[['Defense']] = data frame  
print(type(data_frame))
```

```
<class 'pandas.core.series.Series'>  
<class 'pandas.core.frame.DataFrame'>
```

Before continue with pandas, we need to learn **logic, control flow** and **filtering**.

Comparison operator: ==, <, >, <=

Boolean operators: and, or ,not

Filtering pandas

```
In [17]:  
# Comparison operator  
print(3 > 2)  
print(3!=2)  
  
# Boolean operators  
print(True and False)  
print(True or False)
```

```
True  
True  
False  
True
```

```
In [18]:  
# 1 - Filtering Pandas data frame  
x = data['Defense']>200      # There are only 3 pokemons who have h  
igher defense value than 200  
data[x]
```

Out[18]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gener
224	225	Mega Steelix	Steel	Ground	75	125	230	55	95	30	2
230	231	Shuckle	Bug	Rock	20	10	230	10	230	5	2
333	334	Mega Aggron	Steel	NaN	70	140	230	60	80	50	3

```
In [19]:  
# 2 - Filtering pandas with logical_and  
# There are only 2 pokemons who have higher defence value than 200  
and higher attack value than 100  
data[np.logical_and(data['Defense']>200, data['Attack']>100 )]
```

Out[19]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gener
224	225	Mega Steelix	Steel	Ground	75	125	230	55	95	30	2
333	334	Mega Aggron	Steel	Nan	70	140	230	60	80	50	3

In [20]:

```
# This is also same with previous code line. Therefore we can also
use '&' for filtering.
data[(data['Defense'] > 200) & (data['Attack'] > 100)]
```

Out[20]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gener
224	225	Mega Steelix	Steel	Ground	75	125	230	55	95	30	2
333	334	Mega Aggron	Steel	Nan	70	140	230	60	80	50	3

WHILE and FOR LOOPS

We will learn most basic while and for loops

In [21]:

```
# Stay in loop if condition( i is not equal 5) is true
i = 0
while i != 5 :
    print('i is: ',i)
    i +=1
print(i,' is equal to 5')
```

```
i is: 0
i is: 1
i is: 2
i is: 3
i is: 4
5  is equal to 5
```

In [22]:

```
# Stay in loop if condition( i is not equal 5) is true
lis = [1,2,3,4,5]
for i in lis:
    print('i is: ',i)
print('')
```

```

# Enumerate index and value of list
# index : value = 0:1, 1:2, 2:3, 3:4, 4:5
for index, value in enumerate(lis):
    print(index, " : ", value)
print('')

# For dictionaries
# We can use for loop to achieve key and value of dictionary. We learned key and value at dictionary part.
dictionary = {'spain':'madrid','france':'paris'}
for key,value in dictionary.items():
    print(key, " : ", value)
print('')

# For pandas we can achieve index and value
for index,value in data[['Attack']][0:1].iterrows():
    print(index, " : ", value)

```

```

i is: 1
i is: 2
i is: 3
i is: 4
i is: 5

0 : 1
1 : 2
2 : 3
3 : 4
4 : 5

spain : madrid
france : paris

0 : Attack    49
Name: 0, dtype: int64

```

In this part, you learn:

- how to import csv file
- plotting line, scatter and histogram
- basic dictionary features
- basic pandas features like filtering that is actually something always used and main for being data scientist
- While and for loops

2. PYTHON DATA SCIENCE TOOLBOX

USER DEFINED FUNCTION

What we need to know about functions:

- docstrings: documentation for functions. Example:

```
for f():
    """This is docstring for documentation of function f"""

```
- tuple: sequence of immutable python objects.
 can't modify values
 tuple uses parenthesis like tuple = (1,2,3)
 unpack tuple into several variables like a,b,c = tuple

In [23]:

```
# example of what we learn above
def tuple_ex():
    """ return defined t tuple"""
    t = (1,2,3)
    return t
a,b,c = tuple_ex()
print(a,b,c)
```

1 2 3

SCOPE

What we need to know about scope:

- global: defined main body in script
- local: defined in a function
- built in scope: names in predefined built in scope module such as print, len

Lets make some basic examples

In [24]:

```
# guess print what
x = 2
def f():
    x = 3
    return x
print(x)      # x = 2 global scope
print(f())    # x = 3 local scope
```

2

3

In [25]:

```
# What if there is no local scope
```

```

x = 5
def f():
    y = 2*x      # there is no local scope x
    return y
print(f())      # it uses global scope x
# First local scopesearched, then global scope searched, if two of
# them cannot be found lastly built in scope searched.

```

10

In [26]:

```

# How can we learn what is built in scope
import builtins
dir(builtins)

```

Out[26]:

```

['ArithmeticError',
 'AssertionError',
 'AttributeError',
 'BaseException',
 'BlockingIOError',
 'BrokenPipeError',
 'BufferError',
 'BytesWarning',
 'ChildProcessError',
 'ConnectionAbortedError',
 'ConnectionError',
 'ConnectionRefusedError',
 'ConnectionResetError',
 'DeprecationWarning',
 'EOFError',
 'Ellipsis',
 'EnvironmentError',
 'Exception',
 'False',
 'FileExistsError',
 'FileNotFoundException',
 'FloatingPointError',
 'FutureWarning',
 'GeneratorExit',
 'IOError',
 'ImportError',
 'ImportWarning',
 'IndentationError',
 'IndexError',
 'InterruptedError',
 'IsADirectoryError',
 'KeyError',
 'KeyboardInterrupt',
 'LookupError',
 'MemoryError',
 'ModuleNotFoundError',
 'NameError',
 'None',
 'NotADirectoryError']

```

```
NOTADIRECTORYERROR',
'NotImplemented',
'NotImplementedError',
'OSError',
'OverflowError',
'PendingDeprecationWarning',
'PermissionError',
'ProcessLookupError',
'RecursionError',
'ReferenceError',
'ResourceWarning',
'RuntimetypeError',
'RuntimetypeWarning',
'StopAsyncIteration',
'StopIteration',
'SyntaxError',
'SyntaxWarning',
'SystemError',
'SystemExit',
'TabError',
'TimeoutError',
'True',
'TypeError',
'UnboundLocalError',
'UnicodeDecodeError',
'UnicodeEncodeError',
'UnicodeError',
'UnicodeTranslateError',
'UnicodeWarning',
'UserWarning',
'ValueError',
'Warning',
'ZeroDivisionError',
'__IPYTHON__',
'__build_class__',
'__debug__',
'__doc__',
'__import__',
'__loader__',
'__name__',
'__package__',
'__pybind11_internals_v1__',
'__spec__',
'abs',
'all',
'any',
'ascii',
'bin',
'bool',
'bytearray',
'bytes',
'callable',
'chr',
'classmethod',
'compile',
'complex'
```

```
complex',
'copyright',
'credits',
'delattr',
'dict',
'dir',
'display',
'divmod',
'enumerate',
'eval',
'exec',
'filter',
'float',
'format',
'frozenset',
'get_ipython',
'getattr',
'globals',
'hasattr',
'hash',
'help',
'hex',
'id',
'input',
'int',
'isinstance',
'issubclass',
'iter',
'len',
'license',
'list',
'locals',
'map',
'max',
'memoryview',
'min',
'next',
'object',
'oct',
'open',
'ord',
'pow',
'print',
'property',
'range',
'repr',
'reversed',
'round',
'set',
'setattr',
'slice',
'sorted',
'staticmethod',
'str',
'sum',
'super',
```

```
'tuple',
'type',
'vars',
'zip']
```

NESTED FUNCTION

- function inside function.
- There is a LEGB rule that is search local scope, enclosing function, global and built in scopes, respectively.

In [27]:

```
#nested function
def square():
    """ return square of value """
    def add():
        """ add two local variable """
        x = 2
        y = 3
        z = x + y
        return z
    return add()**2
print(square())
```

25

DEFAULT and FLEXIBLE ARGUMENTS

- Default argument example:

```
def f(a, b=1):
    """ b = 1 is default argument """
```

- Flexible argument example:

```
def f(*args):
    """ *args can be one or more """
```

```
def f(**kwargs)
    """ **kwargs is a dictionary """
```

lets write some code to practice

In [28]:

```
# default arguments
def f(a, b = 1, c = 2):
    y = a + b + c
    return y
print(f(5))
# what if we want to change default arguments
print(f(5,4,3))
```

8

12

In [29]:

```
# flexible arguments *args
def f(*args):
    for i in args:
        print(i)
f(1)
print("")
f(1,2,3,4)
# flexible arguments **kwargs that is dictionary
def f(**kwargs):
    """ print key and value of dictionary"""
    for key, value in kwargs.items():           # If you do not understand this part turn for loop part and look at dictionary in for loop
        print(key, " ", value)
f(country = 'spain', capital = 'madrid', population = 123456)
```

1

1

2

3

4

```
country    spain
capital    madrid
population 123456
```

LAMBDA FUNCTION

Faster way of writing function

In [30]:

```
# lambda function
square = lambda x: x**2      # where x is name of argument
print(square(4))
tot = lambda x,y,z: x+y+z    # where x,y,z are names of arguments
print(tot(1,2,3))
```

```
16
6
```

ANONYMOUS FUNCTION

Like lambda function but it can take more than one arguments.

- map(func,seq) : applies a function to all the items in a list

```
In [31]:
number_list = [1,2,3]
y = map(lambda x:x**2,number_list)
print(list(y))
```

```
[1, 4, 9]
```

ITERATORS

- iterable is an object that can return an iterator
- iterable: an object with an associated iter() method
example: list, strings and dictionaries
- iterator: produces next value with next() method

```
In [32]:
# iteration example
name = "ronaldo"
it = iter(name)
print(next(it))    # print next iteration
print(*it)         # print remaining iteration
```

```
r
o n a l d o
```

`zip()`: zip lists

```
In [33]:
# zip example
list1 = [1,2,3,4]
list2 = [5,6,7,8]
z = zip(list1,list2)
print(z)
z_list = list(z)
print(z_list)
```

```
<zip object at 0x7ff3acf05548>
[(1, 5), (2, 6), (3, 7), (4, 8)]
```

In [34]:

```
un_zip = zip(*z_list)
un_list1,un_list2 = list(un_zip) # unzip returns tuple
print(un_list1)
print(un_list2)
print(type(un_list2))
```

```
(1, 2, 3, 4)
(5, 6, 7, 8)
<class 'tuple'>
```

LIST COMPREHENSION

One of the most important topic of this kernel

We use list comprehension for data analysis often.

list comprehension: collapse for loops for building lists into a single line

Ex: num1 = [1,2,3] and we want to make it num2 = [2,3,4]. This can be done with for loop.

However it is unnecessarily long. We can make it one line code that is list comprehension.

In [35]:

```
# Example of list comprehension
num1 = [1,2,3]
num2 = [i + 1 for i in num1 ]
print(num2)
```

```
[2, 3, 4]
```

[i + 1 for i in num1]: list of comprehension

i +1: list comprehension syntax

for i in num1: for loop syntax

i: iterator

num1: iterable object

In [36]:

```
# Conditionals on iterable
num1 = [5,10,15]
num2 = [i**2 if i == 10 else i-5 if i < 7 else i+5 for i in num1]
print(num2)
```

```
[0, 100, 20]
```

In [37]:

```
# lets return pokemon csv and make one more list comprehension exa
```

```
mples
# lets classify pokemons whether they have high or low speed. Our
# threshold is average speed.
threshold = sum(data.Speed)/len(data.Speed)
data["speed_level"] = ["high" if i > threshold else "low" for i
in data.Speed]
data.loc[:10, ["speed_level", "Speed"]] # we will learn loc more de
tailed later
```

Out[37]:

	speed_level	Speed
0	low	45
1	low	60
2	high	80
3	high	80
4	low	65
5	high	80
6	high	100
7	high	100
8	high	100
9	low	43
10	low	58

Up to now, you learn

- User defined function
- Scope
- Nested function
- Default and flexible arguments
- Lambda function
- Anonymous function
- Iterators
- List comprehension

3.CLEANING DATA

DIAGNOSE DATA for CLEANING

We need to diagnose and clean data before exploring.

Unclean data:

- Column name inconsistency like upper-lower case letter or space between words
- missing data
- different language

We will use head, tail, columns, shape and info methods to diagnose data

In [38]:

```
data = pd.read_csv('../input/pokemon.csv')
data.head() # head shows first 5 rows
```

Out[38]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gener
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1
3	4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1
4	5	Charmander	Fire	NaN	39	52	43	60	50	65	1

In [39]:

```
# tail shows last 5 rows
data.tail()
```

Out[39]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Ge
795	796	Diancie	Rock	Fairy	50	100	150	100	150	50	6
796	797	Mega Diancie	Rock	Fairy	50	160	110	160	110	110	6
797	798	Hoopa Confined	Psychic	Ghost	80	110	60	150	130	70	6
798	799	Hoopa Unbound	Psychic	Dark	80	160	60	170	130	80	6
799	800	Volcanion	Fire	Water	80	110	120	130	90	70	6

In [40]:

```
# columns gives column names of features
data.columns
```

Out[40]:

```
Index(['#', 'Name', 'Type 1', 'Type 2', 'HP', 'Attack', 'Defense',
       'Sp. Atk',
       'Sp. Def', 'Speed', 'Generation', 'Legendary'],
      dtype='object')
```

In [41]:

```
# shape gives number of rows and columns in a table
data.shape
```

Out[41]:

```
(800, 12)
```

In [42]:

```
# info gives data type like dataframe, number of sample or row, number of feature or column, feature types and memory usage
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800 entries, 0 to 799
Data columns (total 12 columns):
 #           800 non-null int64
 Name         799 non-null object
 Type 1       800 non-null object
 Type 2       414 non-null object
 HP           800 non-null int64
 Attack        800 non-null int64
 Defense       800 non-null int64
 Sp. Atk       800 non-null int64
 Sp. Def       800 non-null int64
 Speed          800 non-null int64
 Generation    800 non-null int64
 Legendary     800 non-null bool
 dtypes: bool(1), int64(8), object(3)
memory usage: 69.6+ KB
```

EXPLORATORY DATA ANALYSIS

`value_counts()`: Frequency counts

`outliers`: the value that is considerably higher or lower from rest of the data

- Lets say value at 75% is Q3 and value at 25% is Q1.
- Outlier are smaller than $Q1 - 1.5(Q3-Q1)$ and bigger than $Q3 + 1.5(Q3-Q1)$. ($Q3-Q1 = IQR$)
We will use `describe()` method. Describe method includes:
- `count`: number of entries
- `mean`: average of entries
- `std`: standart deviation
- `min`: minimum entry
- `25%`: first quantile
- `50%`: median or second quantile
- `75%`: third quantile
- `max`: maximum entry

What is quantile?

- 1,4,5,6,8,9,11,12,13,14,15,16,17

- The median is the number that is in **middle** of the sequence. In this case it would be 11.
- The lower quartile is the median in between the smallest number and the median i.e. in between 1 and 11, which is 6.
- The upper quartile, you find the median between the median and the largest number i.e. between 11 and 17, which will be 14 according to the question above.

In [43]:

```
# For example lets look frequency of pokemom types
print(data['Type 1'].value_counts(dropna =False)) # if there are
# nan values that also be counted
# As it can be seen below there are 112 water pokemon or 70 grass
# pokemon
```

Water	112
Normal	98
Grass	70
Bug	69
Psychic	57
Fire	52
Electric	44
Rock	44
Ghost	32
Ground	32
Dragon	32
Dark	31
Poison	28
Fighting	27
Steel	27
Ice	24
Fairy	17
Flying	4
Name: Type 1, dtype: int64	

In [44]:

```
# For example max HP is 255 or min defense is 5
data.describe() #ignore null entries
```

Out[44]:

	#	HP	Attack	Defense	Sp. Atk	Sp. Def	
count	800.0000	800.000000	800.000000	800.000000	800.000000	800.000000	800.000000
mean	400.5000	69.258750	79.001250	73.842500	72.820000	71.902500	68.750000
std	231.0844	25.534669	32.457366	31.183501	32.722294	27.828916	29.500000
min	1.0000	1.000000	5.000000	5.000000	10.000000	20.000000	5.000000
25%	200.7500	50.000000	55.000000	50.000000	49.750000	50.000000	45.000000
50%	400.5000	65.000000	75.000000	70.000000	65.000000	70.000000	65.000000
75%	600.2500	80.000000	100.000000	90.000000	95.000000	90.000000	90.000000
max	800.0000	255.000000	190.000000	230.000000	194.000000	230.000000	180.000000

VISUAL EXPLORATORY DATA ANALYSIS

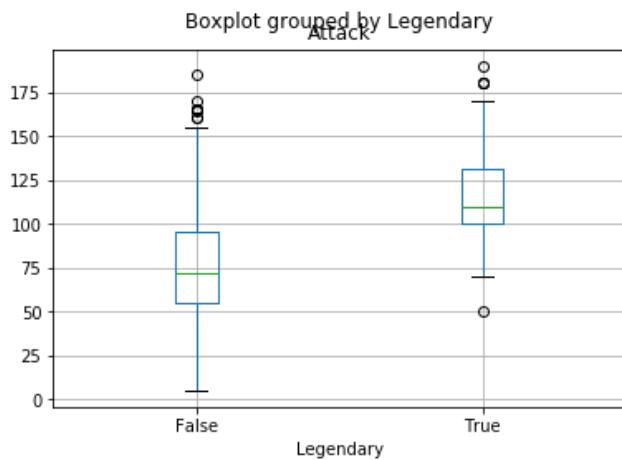
- Box plots: visualize basic statistics like outliers, min/max or quantiles

In [45]:

```
# For example: compare attack of pokemons that are legendary or not
# Black line at top is max
# Blue line at top is 75%
# Red line is median (50%)
# Blue line at bottom is 25%
# Black line at bottom is min
# There are no outliers
data.boxplot(column='Attack', by = 'Legendary')
```

Out[45]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff3acf97940>
```



TIDY DATA

We tidy data with melt(). Describing melt is confusing. Therefore lets make example to understand it.

In [46]:

```
# Firstly I create new data from pokemons data to explain melt more easily.
data_new = data.head()      # I only take 5 rows into new data
data_new
```

Out[46]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gener
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1

#	#	Name	Type	Class	Poison	HP	Attack	Defense	HP	Attack	Defense	#
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1	
3	4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1	
4	5	Charmander	Fire	NaN	39	52	43	60	50	65	1	

In [47]:

```
# lets melt
# id_vars = what we do not wish to melt
# value_vars = what we want to melt
melted = pd.melt(frame=data_new,id_vars = 'Name', value_vars= ['Attack','Defense'])
melted
```

Out[47]:

	Name	variable	value
0	Bulbasaur	Attack	49
1	Ivysaur	Attack	62
2	Venusaur	Attack	82
3	Mega Venusaur	Attack	100
4	Charmander	Attack	52
5	Bulbasaur	Defense	49
6	Ivysaur	Defense	63
7	Venusaur	Defense	83
8	Mega Venusaur	Defense	123
9	Charmander	Defense	43

PIVOTING DATA

Reverse of melting.

In [48]:

```
# Index is name
# I want to make that columns are variable
# Finally values in columns are value
melted.pivot(index = 'Name', columns = 'variable',values='value')
```

Out[48]:

variable	Attack	Defense
Name		
Bulbasaur	49	49
Charmander	52	43
Ivysaur	62	63

Mega Venusaur	100	123
Venusaur	82	83

CONCATENATING DATA

We can concatenate two dataframe

In [49]:

```
# Firstly lets create 2 data frame
data1 = data.head()
data2= data.tail()
conc_data_row = pd.concat([data1,data2],axis =0,ignore_index =True)
# axis = 0 : adds dataframes in row
conc_data_row
```

Out[49]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Ge
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1
3	4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1
4	5	Charmander	Fire	NaN	39	52	43	60	50	65	1
5	796	Diancie	Rock	Fairy	50	100	150	100	150	50	6
6	797	Mega Diancie	Rock	Fairy	50	160	110	160	110	110	6
7	798	Hoopa Confined	Psychic	Ghost	80	110	60	150	130	70	6
8	799	Hoopa Unbound	Psychic	Dark	80	160	60	170	130	80	6
9	800	Volcanion	Fire	Water	80	110	120	130	90	70	6

In [50]:

```
data1 = data['Attack'].head()
data2= data['Defense'].head()
conc_data_col = pd.concat([data1,data2],axis =1) # axis = 0 : adds dataframes in row
conc_data_col
```

Out[50]:

	Attack	Defense
0	49	49
1	62	63

2	82	83
3	100	123
4	52	43

DATA TYPES

There are 5 basic data types: object(string), boolean, integer, float and categorical.

We can make conversion data types like from str to categorical or from int to float

Why is category important:

- make dataframe smaller in memory
- can be utilized for analysis especially for sklearn (we will learn later)

```
In [51]:  
      data.dtypes
```

```
Out[51]:  
#          int64  
Name      object  
Type 1    object  
Type 2    object  
HP        int64  
Attack    int64  
Defense   int64  
Sp. Atk   int64  
Sp. Def   int64  
Speed     int64  
Generation int64  
Legendary  bool  
dtype: object
```

```
In [52]:  
# lets convert object(str) to categorical and int to float.  
data['Type 1'] = data['Type 1'].astype('category')  
data['Speed'] = data['Speed'].astype('float')
```

```
In [53]:  
# As you can see Type 1 is converted from object to categorical  
# And Speed ,s converted from int to float  
data.dtypes
```

```
Out[53]:  
#          int64  
Name      object  
Type 1    category  
Type 2    object  
HP        int64  
Attack    int64  
Defense   int64
```

```
Sp. Atk      int64
Sp. Def     int64
Speed       float64
Generation   int64
Legendary    bool
dtype: object
```

MISSING DATA and TESTING WITH ASSERT

If we encounter with missing data, what we can do:

- leave as is
- drop them with dropna()
- fill missing value with fillna()
- fill missing values with test statistics like mean

Assert statement: check that you can turn on or turn off when you are done with your testing of the program

In [54]:

```
# Lets look at does pokemon data have nan value
# As you can see there are 800 entries. However Type 2 has 414 non
>null object so it has 386 null object.
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800 entries, 0 to 799
Data columns (total 12 columns):
 #           800 non-null int64
 Name        799 non-null object
 Type 1      800 non-null category
 Type 2      414 non-null object
 HP          800 non-null int64
 Attack      800 non-null int64
 Defense     800 non-null int64
 Sp. Atk     800 non-null int64
 Sp. Def     800 non-null int64
 Speed       800 non-null float64
 Generation   800 non-null int64
 Legendary    800 non-null bool
 dtypes: bool(1), category(1), float64(1), int64(7), object(2)
 memory usage: 64.9+ KB
```

In [55]:

```
# Lets chech Type 2
data["Type 2"].value_counts(dropna =False)
# As you can see, there are 386 NAN value
```

Out[55]:

NaN	386
Flying	97
Ground	35

```

Poison      34
Psychic    33
Fighting   26
Grass       25
Fairy       23
Steel        22
Dark        20
Dragon      18
Ice         14
Ghost        14
Rock        14
Water        14
Fire         12
Electric     6
Normal       4
Bug          3
Name: Type 2, dtype: int64

```

In [56]:

```

# Lets drop nan values
data1=data  # also we will use data to fill missing value so I assign it to data1 variable
data1["Type 2"].dropna(inplace = True) # inplace = True means we do not assign it to new variable. Changes automatically assigned to data
# So does it work ?

```

In [57]:

```

# Lets check with assert statement
# Assert statement:
assert 1==1 # return nothing because it is true

```

In [58]:

```

# In order to run all code, we need to make this line comment
# assert 1==2 # return error because it is false

```

In [59]:

```

assert data['Type 2'].notnull().all() # returns nothing because we drop nan values

```

In [60]:

```

data["Type 2"].fillna('empty',inplace = True)

```

In [61]:

```

assert data['Type 2'].notnull().all() # returns nothing because we do not have nan values

```

In [62]:

```

# # With assert statement we can check a lot of thing. For example
# assert data.columns[1] == 'Name'
# assert data.Speed.dtypes == np.int

```

In this part, you learn:

- Diagnose data for cleaning
- Exploratory data analysis
- Visual exploratory data analysis
- Tidy data
- Pivoting data
- Concatenating data
- Data types
- Missing data and testing with assert

4. PANDAS FOUNDATION

REVIEW of PANDAS

As you notice, I do not give all idea in a same time. Although, we learn some basics of pandas, we will go deeper in pandas.

- single column = series
- NaN = not a number
- dataframe.values = numpy

BUILDING DATA FRAMES FROM SCRATCH

- We can build data frames from csv as we did earlier.
- Also we can build dataframe from dictionaries
 - zip() method: This function returns a list of tuples, where the i-th tuple contains the i-th element from each of the argument sequences or iterables.
- Adding new column
- Broadcasting: Create new column and assign a value to entire column

In [63]:

```
# data frames from dictionary
country = ["Spain", "France"]
population = ["11", "12"]
list_label = ["country", "population"]
list_col = [country, population]
zipped = list(zip(list_label, list_col))
data_dict = dict(zipped)
df = pd.DataFrame(data_dict)
```

df

Out[63]:

	country	population
0	Spain	11
1	France	12

In [64]:

```
# Add new columns
df["capital"] = ["madrid", "paris"]
df
```

Out[64]:

	country	population	capital
0	Spain	11	madrid
1	France	12	paris

In [65]:

```
# Broadcasting
df["income"] = 0 #Broadcasting entire column
df
```

Out[65]:

	country	population	capital	income
0	Spain	11	madrid	0
1	France	12	paris	0

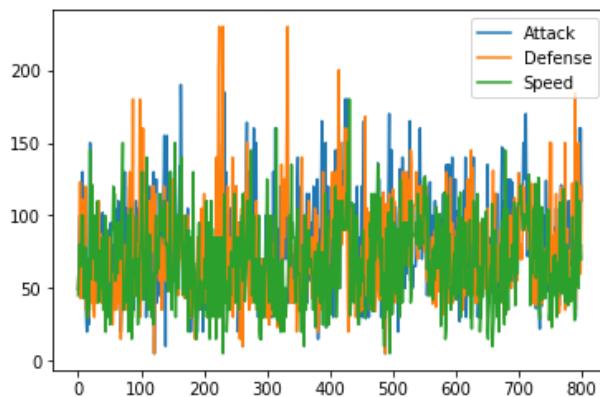
VISUAL EXPLORATORY DATA ANALYSIS

- Plot
- Subplot
- Histogram:
 - bins: number of bins
 - range(table): min and max values of bins
 - normed(boolean): normalize or not
 - cumulative(boolean): compute cumulative distribution

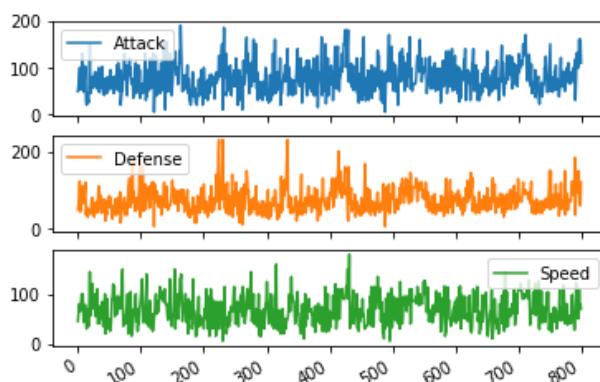
In [66]:

```
# Plotting all data
data1 = data.loc[:, ["Attack", "Defense", "Speed"]]
data1.plot()
# it is confusing
```

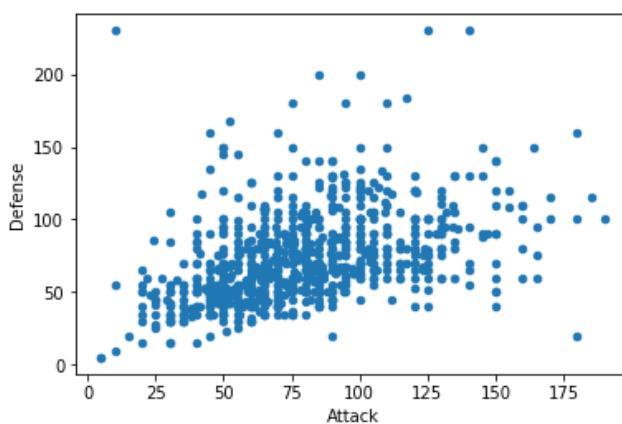
```
Out[66]:  
<matplotlib.axes._subplots.AxesSubplot at 0x7ff3aceade10>
```



```
In [67]:  
# subplots  
data1.plot(subplots = True)  
plt.show()
```



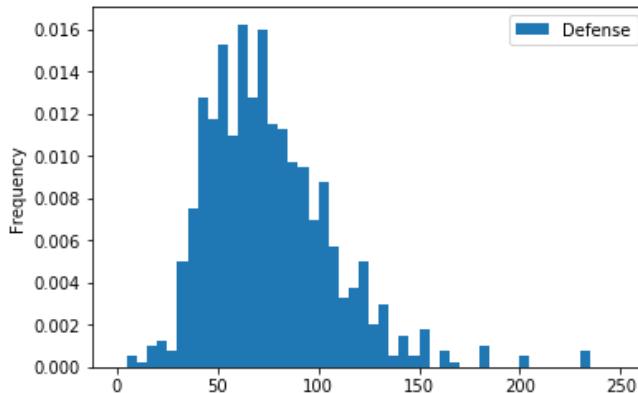
```
In [68]:  
# scatter plot  
data1.plot(kind = "scatter",x="Attack",y = "Defense")  
plt.show()
```



```
In [69]:  
# hist + norm
```

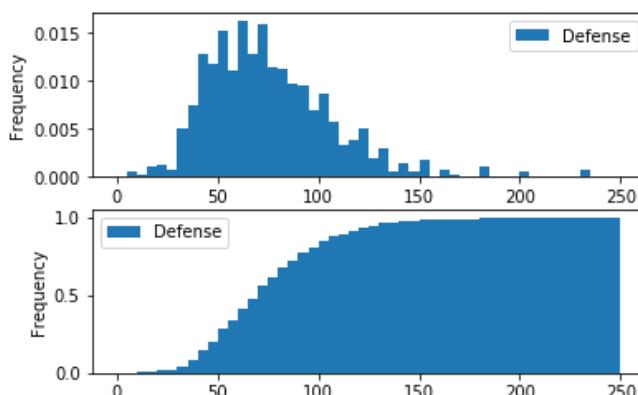
```
# hist plot
data1.plot(kind = "hist",y = "Defense",bins = 50,range= (0,250),normed = True)
```

Out[69]:
<matplotlib.axes._subplots.AxesSubplot at 0x7ff3acccdb70>



In [70]:
histogram subplot with non cumulative and cumulative
fig, axes = plt.subplots(nrows=2,ncols=1)
data1.plot(kind = "hist",y = "Defense",bins = 50,range= (0,250),normed = True,ax = axes[0])
data1.plot(kind = "hist",y = "Defense",bins = 50,range= (0,250),normed = True,ax = axes[1],cumulative = True)
plt.savefig('graph.png')
plt

Out[70]:
<module 'matplotlib.pyplot' from '/opt/conda/lib/python3.6/site-packages/matplotlib/pyplot.py'>



STATISTICAL EXPLORATORY DATA ANALYSIS

I already explained it at previous parts. However lets look at one more time.

- count: number of entries
- mean: average of entries

and so on...

- std: standard deviation
- min: minimum entry
- 25%: first quantile
- 50%: median or second quantile
- 75%: third quantile
- max: maximum entry

In [71]:

```
data.describe()
```

Out[71]:

	#	HP	Attack	Defense	Sp. Atk	Sp. Def	
count	800.0000	800.000000	800.000000	800.000000	800.000000	800.000000	800.000000
mean	400.5000	69.258750	79.001250	73.842500	72.820000	71.902500	68.500000
std	231.0844	25.534669	32.457366	31.183501	32.722294	27.828916	29.500000
min	1.0000	1.000000	5.000000	5.000000	10.000000	20.000000	5.000000
25%	200.7500	50.000000	55.000000	50.000000	49.750000	50.000000	45.000000
50%	400.5000	65.000000	75.000000	70.000000	65.000000	70.000000	65.000000
75%	600.2500	80.000000	100.000000	90.000000	95.000000	90.000000	90.000000
max	800.0000	255.000000	190.000000	230.000000	194.000000	230.000000	180.000000

INDEXING PANDAS TIME SERIES

- datetime = object
- parse_dates(boolean): Transform date to ISO 8601 (yyyy-mm-dd hh:mm:ss) format

In [72]:

```
time_list = ["1992-03-08", "1992-04-12"]
print(type(time_list[1])) # As you can see date is string
# however we want it to be datetime object
datetime_object = pd.to_datetime(time_list)
print(type(datetime_object))
```

```
<class 'str'>
<class 'pandas.core.indexes.datetimes.DatetimeIndex'>
```

In [73]:

```
# close warning
import warnings
warnings.filterwarnings("ignore")
# In order to practice lets take head of pokemon data and add it a
# time list
data2 = data.head()
date_list = ["1992-01-10", "1992-02-10", "1992-03-10", "1993-03-15",
"1993-03-16"]
```

```

datetime_object = pd.to_datetime(date_list)
data2["date"] = datetime_object
# lets make date as index
data2= data2.set_index("date")
data2

```

Out[73]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	G
date											
1992-01-10	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45.0	1
1992-02-10	2	Ivysaur	Grass	Poison	60	62	63	80	80	60.0	1
1992-03-10	3	Venusaur	Grass	Poison	80	82	83	100	100	80.0	1
1993-03-15	4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80.0	1
1993-03-16	5	Charmander	Fire	NaN	39	52	43	60	50	65.0	1

In [74]:

```

# Now we can select according to our date index
print(data2.loc["1993-03-16"])
print(data2.loc["1992-03-10":"1993-03-16"])

```

```

#          5
Name      Charmander
Type 1     Fire
Type 2     NaN
HP         39
Attack     52
Defense    43
Sp. Atk    60
Sp. Def    50
Speed      65
Generation 1
Legendary  False
Name: 1993-03-16 00:00:00, dtype: object
#          Name Type 1 ... Speed Generation
Legendray
date           ...
1992-03-10  3   Venusaur Grass ... 80.0       1
             False
1993-03-15  4   Mega Venusaur Grass ... 80.0       1
             False
1993-03-16  5   Charmander Fire  ... 65.0       1
             False

[3 rows x 12 columns]

```

RESAMPLING PANDAS TIME SERIES

- Resampling: statistical method over different time intervals
 - Needs string to specify frequency like "M" = month or "A" = year
- Downsampling: reduce date time rows to slower frequency like from daily to weekly
- Upsampling: increase date time rows to faster frequency like from daily to hourly
- Interpolate: Interpolate values according to different methods like 'linear', 'time' or index'
 - [\(https://pandas.pydata.org/pandas-docs/stable/generated/pandas.Series.interpolate.html\)](https://pandas.pydata.org/pandas-docs/stable/generated/pandas.Series.interpolate.html)

In [75]:

```
# We will use data2 that we create at previous part
data2.resample("A").mean()
```

Out[75]:

	#	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gen
date								
1992-12-31	2.0	61.666667	64.333333	65.0	81.666667	81.666667	61.666667	1.0
1993-12-31	4.5	59.500000	76.000000	83.0	91.000000	85.000000	72.500000	1.0

In [76]:

```
# Lets resample with month
data2.resample("M").mean()
# As you can see there are a lot of nan because data2 does not include all months
```

Out[76]:

	#	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legendary
date									
1992-01-31	1.0	45.0	49.0	49.0	65.0	65.0	45.0	1.0	0.0
1992-02-29	2.0	60.0	62.0	63.0	80.0	80.0	60.0	1.0	0.0
1992-03-31	3.0	80.0	82.0	83.0	100.0	100.0	80.0	1.0	0.0
1992-04-30	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-05-31	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-06-30	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Ub-30									
1992-07-31	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-08-31	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-09-30	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-10-31	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-11-30	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1992-12-31	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1993-01-31	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1993-02-28	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1993-03-31	4.5	59.5	76.0	83.0	91.0	85.0	72.5	1.0	0.0

In [77]:

```
# In real life (data is real. Not created from us like data2) we can solve this problem with interpolate
# We can interpolate from first value
data2.resample("M").first().interpolate("linear")
```

Out[77]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk
date								
1992-01-31	1.000000	Bulbasaur	Grass	Poison	45.0	49.0	49.000000	65.000000
1992-02-29	2.000000	Ivysaur	Grass	Poison	60.0	62.0	63.000000	80.000000
1992-03-31	3.000000	Venusaur	Grass	Poison	80.0	82.0	83.000000	100.000000
1992-04-30	3.083333	NaN	NaN	NaN	80.0	83.5	86.333333	101.833333
1992-05-31	3.166667	NaN	NaN	NaN	80.0	85.0	89.666667	103.666667
1992-06-30	3.250000	NaN	NaN	NaN	80.0	86.5	93.000000	105.500000
1992-07-31	3.333333	NaN	NaN	NaN	80.0	88.0	96.333333	107.333333
1992-08-31	3.416667	NaN	NaN	NaN	80.0	89.5	99.666667	109.166667
1992-09-30	3.500000	NaN	NaN	NaN	80.0	91.0	103.000000	111.000000
1992-10-31	3.583333	NaN	NaN	NaN	80.0	92.5	106.333333	112.833333

1992-11-30	3.666667	NaN	NaN	NaN	80.0	94.0	109.666667	114.666667
1992-12-31	3.750000	NaN	NaN	NaN	80.0	95.5	113.000000	116.500000
1993-01-31	3.833333	NaN	NaN	NaN	80.0	97.0	116.333333	118.333333
1993-02-28	3.916667	NaN	NaN	NaN	80.0	98.5	119.666667	120.166667
1993-03-31	4.000000	Mega Venusaur	Grass	Poison	80.0	100.0	123.000000	122.000000

In [78]:

```
# Or we can interpolate with mean()
data2.resample("M").mean().interpolate("linear")
```

Out[78]:

	#	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Leg
date									
1992-01-31	1.000	45.000000	49.0	49.0	65.00	65.00	45.000	1.0	0.0
1992-02-29	2.000	60.000000	62.0	63.0	80.00	80.00	60.000	1.0	0.0
1992-03-31	3.000	80.000000	82.0	83.0	100.00	100.00	80.000	1.0	0.0
1992-04-30	3.125	78.291667	81.5	83.0	99.25	98.75	79.375	1.0	0.0
1992-05-31	3.250	76.583333	81.0	83.0	98.50	97.50	78.750	1.0	0.0
1992-06-30	3.375	74.875000	80.5	83.0	97.75	96.25	78.125	1.0	0.0
1992-07-31	3.500	73.166667	80.0	83.0	97.00	95.00	77.500	1.0	0.0
1992-08-31	3.625	71.458333	79.5	83.0	96.25	93.75	76.875	1.0	0.0
1992-09-30	3.750	69.750000	79.0	83.0	95.50	92.50	76.250	1.0	0.0
1992-10-31	3.875	68.041667	78.5	83.0	94.75	91.25	75.625	1.0	0.0
1992-11-30	4.000	66.333333	78.0	83.0	94.00	90.00	75.000	1.0	0.0
1992-12-31	4.125	64.625000	77.5	83.0	93.25	88.75	74.375	1.0	0.0
1993-01-31	4.250	62.916667	77.0	83.0	92.50	87.50	73.750	1.0	0.0
1993-02-28	4.375	61.208333	76.5	83.0	91.75	86.25	73.125	1.0	0.0
1993-03-31	4.500	59.500000	76.0	83.0	91.00	85.00	72.500	1.0	0.0

MANIPULATING DATA FRAMES WITH PANDAS

INDEXING DATA FRAMES

- Indexing using square brackets
- Using column attribute and row label
- Using loc accessor
- Selecting only some columns

In [79]:

```
# read data
data = pd.read_csv('../input/pokemon.csv')
data= data.set_index("#")
data.head()
```

Out[79]:

	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation
#										
1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
3	Venusaur	Grass	Poison	80	82	83	100	100	80	1
4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1
5	Charmander	Fire	NaN	39	52	43	60	50	65	1

In [80]:

```
# indexing using square brackets
data["HP"][1]
```

Out[80]:

45

In [81]:

```
# using column attribute and row label
data.HP[1]
```

Out[81]:

45

In [82]:

```
# using loc accessor
```

```
# Using loc accessor  
data.loc[1, ["HP"]]
```

```
Out[82]:  
HP    45  
Name: 1, dtype: object
```

```
In [83]:  
# Selecting only some columns  
data[["HP", "Attack"]]
```

```
Out[83]:
```

	HP	Attack
#		
1	45	49
2	60	62
3	80	82
4	80	100
5	39	52
6	58	64
7	78	84
8	78	130
9	78	104
10	44	48
11	59	63
12	79	83
13	79	103
14	45	30
15	50	20
16	60	45
17	40	35
18	45	25
19	65	90
20	65	150
21	40	45
22	63	60
23	83	80
24	83	80
25	30	56
26	55	81
27	40	60
28	65	90
29	35	60
30	60	85
...
771	95	65
772	70	80

	172	178	92
773	67	58	
774	50	50	
775	45	50	
776	68	75	
777	90	100	
778	57	80	
779	43	70	
780	85	110	
781	49	66	
782	44	66	
783	54	66	
784	59	66	
785	65	90	
786	55	85	
787	75	95	
788	85	100	
789	55	69	
790	95	117	
791	40	30	
792	85	70	
793	126	131	
794	126	131	
795	108	100	
796	50	100	
797	50	160	
798	80	110	
799	80	160	
800	80	110	

800 rows × 2 columns

SLICING DATA FRAME

- Difference between selecting columns
 - Series and data frames
- Slicing and indexing series
- Reverse slicing
- From something to end

In [84]:

```
# Difference between selecting columns: series and dataframes
```

```
print(type(data["HP"]))      # series
print(type(data[["HP"]]))    # data frames
```

```
<class 'pandas.core.series.Series'>
<class 'pandas.core.frame.DataFrame'>
```

In [85]:

```
# Slicing and indexing series
data.loc[1:10, "HP":"Defense"]  # 10 and "Defense" are inclusive
```

Out[85]:

	HP	Attack	Defense
#			
1	45	49	49
2	60	62	63
3	80	82	83
4	80	100	123
5	39	52	43
6	58	64	58
7	78	84	78
8	78	130	111
9	78	104	78
10	44	48	65

In [86]:

```
# Reverse slicing
data.loc[10:1:-1, "HP":"Defense"]
```

Out[86]:

	HP	Attack	Defense
#			
10	44	48	65
9	78	104	78
8	78	130	111
7	78	84	78
6	58	64	58
5	39	52	43
4	80	100	123
3	80	82	83
2	60	62	63
1	45	49	49

In [87]:

```
# From something to end
```

```
data.loc[1:10, "Speed":]
```

Out[87]:

	Speed	Generation	Legendary
#			
1	45	1	False
2	60	1	False
3	80	1	False
4	80	1	False
5	65	1	False
6	80	1	False
7	100	1	False
8	100	1	False
9	100	1	False
10	43	1	False

FILTERING DATA FRAMES

Creating boolean series Combining filters Filtering column based others

In [88]:

```
# Creating boolean series
boolean = data.HP > 200
data[boolean]
```

Out[88]:

	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation
#										
122	Chansey	Normal	NaN	250	5	5	35	105	50	1
262	Blissey	Normal	NaN	255	10	10	75	135	55	2

In [89]:

```
# Combining filters
first_filter = data.HP > 150
second_filter = data.Speed > 35
data[first_filter & second_filter]
```

Out[89]:

	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation
#										
122	Chansey	Normal	NaN	250	5	5	35	105	50	1

ID	Category	Type	Attack	Defense	HP	Speed	Attack	Defense	HP	Speed
262	Blissey	Normal	NaN	255	10	10	75	135	55	2
352	Wailord	Water	NaN	170	90	45	90	45	60	3
656	Alomomola	Water	NaN	165	75	80	40	45	65	5

In [90]:

```
# Filtering column based others
data.HP[data.Speed<15]
```

Out[90]:

```
#  
231    20  
360    45  
487    50  
496    135  
659    44  
Name: HP, dtype: int64
```

TRANSFORMING DATA

- Plain python functions
- Lambda function: to apply arbitrary python function to every element
- Defining column using other columns

In [91]:

```
# Plain python functions
def div(n):
    return n/2
data.HP.apply(div)
```

Out[91]:

```
#  
1    22.5  
2    30.0  
3    40.0  
4    40.0  
5    19.5  
6    29.0  
7    39.0  
8    39.0  
9    39.0  
10   22.0  
11   29.5  
12   39.5  
13   39.5  
14   22.5  
15   25.0  
16   30.0  
17   20.0
```

```
18    22.5
19    32.5
20    32.5
21    20.0
22    31.5
23    41.5
24    41.5
25    15.0
26    27.5
27    20.0
28    32.5
29    17.5
30    30.0
...
771   47.5
772   39.0
773   33.5
774   25.0
775   22.5
776   34.0
777   45.0
778   28.5
779   21.5
780   42.5
781   24.5
782   22.0
783   27.0
784   29.5
785   32.5
786   27.5
787   37.5
788   42.5
789   27.5
790   47.5
791   20.0
792   42.5
793   63.0
794   63.0
795   54.0
796   25.0
797   25.0
798   40.0
799   40.0
800   40.0
Name: HP, Length: 800, dtype: float64
```

In [92]:

```
# Or we can use lambda function
data.HP.apply(lambda n : n/2)
```

Out[92]:

```
#
1    22.5
2    30.0
3    40.0
4    10.0
```

4	40.0
5	19.5
6	29.0
7	39.0
8	39.0
9	39.0
10	22.0
11	29.5
12	39.5
13	39.5
14	22.5
15	25.0
16	30.0
17	20.0
18	22.5
19	32.5
20	32.5
21	20.0
22	31.5
23	41.5
24	41.5
25	15.0
26	27.5
27	20.0
28	32.5
29	17.5
30	30.0
	...
771	47.5
772	39.0
773	33.5
774	25.0
775	22.5
776	34.0
777	45.0
778	28.5
779	21.5
780	42.5
781	24.5
782	22.0
783	27.0
784	29.5
785	32.5
786	27.5
787	37.5
788	42.5
789	27.5
790	47.5
791	20.0
792	42.5
793	63.0
794	63.0
795	54.0
796	25.0
797	25.0
798	10.0

```
, 799    40.0
800    40.0
Name: HP, Length: 800, dtype: float64
```

In [93]:

```
# Defining column using other columns
data["total_power"] = data.Attack + data.Defense
data.head()
```

Out[93]:

	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation
#										
1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
3	Venusaur	Grass	Poison	80	82	83	100	100	80	1
4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1
5	Charmander	Fire	NaN	39	52	43	60	50	65	1

INDEX OBJECTS AND LABELED DATA

index: sequence of label

In [94]:

```
# our index name is this:
print(data.index.name)
# lets change it
data.index.name = "index_name"
data.head()
```

#

Out[94]:

	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed
index_name									
1	Bulbasaur	Grass	Poison	45	49	49	65	65	45
2	Ivysaur	Grass	Poison	60	62	63	80	80	60
3	Venusaur	Grass	Poison	80	82	83	100	100	80
4	Mega Venusaur	Grass	Poison	80	100	123	122	120	80
5	Charmander	Fire	NaN	39	52	43	60	50	65

In [95]:

```
# Overwrite index
# if we want to modify index we need to change all of them.
data.head()

# first copy of our data to data3 then change index
data3 = data.copy()

# lets make index start from 100. It is not remarkable change but
# it is just example
data3.index = range(100, 900, 1)
data3.head()
```

Out[95]:

	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generat
100	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
101	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
102	Venusaur	Grass	Poison	80	82	83	100	100	80	1
103	Mega Venusaur	Grass	Poison	80	100	123	122	120	80	1
104	Charmander	Fire	NaN	39	52	43	60	50	65	1

In [96]:

```
# We can make one of the column as index. I actually did it at the
# beginning of manipulating data frames with pandas section
# It was like this
# data= data.set_index("#")
# also you can use
# data.index = data["#"]
```

HIERARCHICAL INDEXING

- Setting indexing

In [97]:

```
# lets read data frame one more time to start from beginning
data = pd.read_csv('../input/pokemon.csv')
data.head()

# As you can see there is index. However we want to set one or more
# column to be index
```

Out[97]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gener
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1

#	Type 1	Type 2	Name	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gen	
3	Grass	Poison	Bulbasaur	45	49	49	65	65	45	1	
4	Fire	NaN	Charmander	39	52	43	60	50	65	1	

In [98]:

```
# Setting index : type 1 is outer type 2 is inner index
data1 = data.set_index(["Type 1", "Type 2"])
data1.head(100)
# data1.loc["Fire", "Flying"] # howw to use indexes
```

Out[98]:

Type 1	Type 2	#	Name	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Gen	
Grass	Poison	1	Bulbasaur	45	49	49	65	65	45	1	
	Poison	2	Ivysaur	60	62	63	80	80	60	1	
	Poison	3	Venusaur	80	82	83	100	100	80	1	
	Poison	4	Mega Venusaur	80	100	123	122	120	80	1	
Fire	NaN	5	Charmander	39	52	43	60	50	65	1	
	NaN	6	Charmeleon	58	64	58	80	65	80	1	
	Flying	7	Charizard	78	84	78	109	85	100	1	
	Dragon	8	Mega Charizard X	78	130	111	130	85	100	1	
	Flying	9	Mega Charizard Y	78	104	78	159	115	100	1	
Water	NaN	10	Squirtle	44	48	65	50	64	43	1	
	NaN	11	Wartortle	59	63	80	65	80	58	1	
	NaN	12	Blastoise	79	83	100	85	105	78	1	
	NaN	13	Mega Blastoise	79	103	120	135	115	78	1	
Bug	NaN	14	Caterpie	45	30	35	20	20	45	1	
	NaN	15	Metapod	50	20	55	25	25	30	1	
	Flying	16	Butterfree	60	45	50	90	80	70	1	
	Poison	17	Weedle	40	35	30	20	20	50	1	
	Poison	18	Kakuna	45	25	50	25	25	35	1	
	Poison	19	Beedrill	65	90	40	45	80	75	1	
	Poison	20	Mega Beedrill	65	150	40	15	80	145	1	
Normal	Flying	21	Pidgey	40	45	40	35	35	56	1	
	Flying	22	Pidgeotto	63	60	55	50	50	71	1	
	Flying	23	Pidgeot	83	80	75	70	70	101	1	
	Flying	24	Mega Pidgeot	83	80	80	135	80	121	1	
	NaN	25	Rattata	30	56	35	25	35	72	1	
	NaN	26	Raticate	55	81	60	50	70	97	1	

	Flying	27	Spearow	40	60	30	31	31	70	1
	Flying	28	Fearow	65	90	65	61	61	100	1
Poison	NaN	29	Ekans	35	60	44	40	54	55	1
	NaN	30	Arbok	60	85	69	65	79	80	1
...
Psychic	NaN	71	Alakazam	55	50	45	135	95	120	1
	NaN	72	Mega Alakazam	55	50	65	175	95	150	1
Fighting	NaN	73	Machop	70	80	50	35	35	35	1
	NaN	74	Machoke	80	100	70	50	60	45	1
	NaN	75	Machamp	90	130	80	65	85	55	1
Grass	Poison	76	Bellsprout	50	75	35	70	30	40	1
	Poison	77	Weepinbell	65	90	50	85	45	55	1
	Poison	78	Vicreebel	80	105	65	100	70	70	1
Water	Poison	79	Tentacool	40	40	35	50	100	70	1
	Poison	80	Tentacruel	80	70	65	80	120	100	1
Rock	Ground	81	Geodude	40	80	100	30	30	20	1
	Ground	82	Graveler	55	95	115	45	45	35	1
	Ground	83	Golem	80	120	130	55	65	45	1
Fire	NaN	84	Ponyta	50	85	55	65	65	90	1
	NaN	85	Rapidash	65	100	70	80	80	105	1
Water	Psychic	86	Slowpoke	90	65	65	40	40	15	1
	Psychic	87	Slowbro	95	75	110	100	80	30	1
	Psychic	88	Mega Slowbro	95	75	180	130	80	30	1
Electric	Steel	89	Magnemite	25	35	70	95	55	45	1
	Steel	90	Magneton	50	60	95	120	70	70	1
Normal	Flying	91	Farfetch'd	52	65	55	58	62	60	1
	Flying	92	Doduo	35	85	45	35	35	75	1
	Flying	93	Dodrio	60	110	70	60	60	100	1
Water	NaN	94	Seel	65	45	55	45	70	45	1
	Ice	95	Dewgong	90	70	80	70	95	70	1
Poison	NaN	96	Grimer	80	80	50	40	50	25	1
	NaN	97	Muk	105	105	75	65	100	50	1
Water	NaN	98	Shellder	30	65	100	45	25	40	1
	Ice	99	Cloyster	50	95	180	85	45	70	1
Ghost	Poison	100	Gastly	30	35	30	100	35	80	1

100 rows × 10 columns

PIVOTING DATA FRAMES

- pivoting: reshape tool

In [99]:

```
dic = {"treatment": ["A", "A", "B", "B"], "gender": ["F", "M", "F", "M"],
       "response": [10, 45, 5, 9], "age": [15, 4, 72, 65]}
df = pd.DataFrame(dic)
df
```

Out[99]:

	treatment	gender	response	age
0	A	F	10	15
1	A	M	45	4
2	B	F	5	72
3	B	M	9	65

In [100]:

```
# pivoting
df.pivot(index="treatment", columns = "gender", values="response")
```

Out[100]:

gender	F	M
treatment		
A	10	45
B	5	9

STACKING and UNSTACKING DATAFRAME

- deal with multi label indexes
- level: position of unstacked index
- swaplevel: change inner and outer level index position

In [101]:

```
df1 = df.set_index(["treatment", "gender"])
df1
# lets unstack it
```

Out[101]:

		response	age
treatment	gender		
A	F	10	15
	M	45	4
B	F	5	72

	M	9	65
--	---	---	----

In [102]:

```
# level determines indexes
df1.unstack(level=0)
```

Out[102]:

	response		age	
treatment	A	B	A	B
gender				
F	10	5	15	72
M	45	9	4	65

In [103]:

```
df1.unstack(level=1)
```

Out[103]:

	response		age	
gender	F	M	F	M
treatment				
A	10	45	15	4
B	5	9	72	65

In [104]:

```
# change inner and outer level index position
df2 = df1.swaplevel(0,1)
df2
```

Out[104]:

		response	age
gender	treatment		
F	A	10	15
M	A	45	4
F	B	5	72
M	B	9	65

MELTING DATA FRAMES

- Reverse of pivoting

Did you find this Kernel useful?
Show your appreciation with an upvote

2087



Comments (677)

All Comments ▾

Sort by

Hotness ▾



Click here to enter a comment...



Bulent Siyah • Posted on Latest Version • a month ago • Options • Reply

^ 1 ▼

it is really very useful. Thanks a lot for the effort.. Great notebook for the beginners(like me).



Corpus • Posted on Version 739 • 2 months ago • Options • Reply

^ 2 ▼

Amazing job, Really helpful for beginners like myself. Thank you, keep it up! :3

DATAI

DATAI Kernel Author

• Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

You are welcome



AstroBoy • Posted on Version 734 • 3 months ago • Options • Reply

^ 2 ▼



Excellent work. Thanks for sharing this Kaan. Useful tutorial to know the basics and start exploring.

DATAI

DATAI Kernel Author

• Posted on Version 734 • 3 months ago • Options • Reply

^ 0 ▼

Thank you. I glad it is helpful to you.



Atul Kumar • Posted on Version 628 • 5 months ago • Options • Reply

^ 1 ▼



Thanks, Kaan!! Very Useful for the beginners. Keep up the great work.

DATAI

DATAI Kernel Author

• Posted on Version 629 • 5 months ago • Options • Reply

^ 0 ▼

Thank you Atul. I glad you like it. I will make this type of works with yours support.



Sarthak Mahapatra · Posted on Version 628 · 5 months ago · Options · Reply

^ 1 ▼



Great Notebook for beginners . Thanks :)

DATAI

DATAI Kernel Author

· Posted on Version 629 · 5 months ago · Options · Reply

^ 0 ▼

•••• You are welcome Sarthak



meng · Posted on Version 625 · 5 months ago · Options · Reply

^ 1 ▼



much usefull

DATAI

DATAI Kernel Author

· Posted on Version 629 · 5 months ago · Options · Reply

^ 0 ▼

•••• Thanks :)



Erik Bruun · Posted on Version 628 · 5 months ago · Options · Reply

^ 1 ▼



628 versions, of which the last 623 without changes. Please stop running this for no reason.....

DATAI

DATAI Kernel Author

· Posted on Version 629 · 5 months ago · Options · Reply

^ 5 ▼

•••• Yes there is a reason. When i run, it appears in kaggle docker. I will run whenever i want....



Mark · Posted on Version 586 · 5 months ago · Options · Reply

^ 1 ▼



Thanks Kaan, super useful notebook which has helped put what i've been learning over the last few weeks into practice

DATAI

DATAI Kernel Author

· Posted on Version 588 · 5 months ago · Options · Reply

^ 0 ▼

•••• Thank you for your comment Mark.



vikasfugoki · Posted on Version 577 · 5 months ago · Options · Reply

^ 1 ▼



great notebook for the beginners



Adrián López • Posted on Version 364 • 7 months ago • Options • Reply

^ 3 ▾

I think there is a small error in your in[50] example explanation.

```
conc_data_col = pd.concat([data1,data2],axis =1) # axis = 0 : adds dataframes in row
```

There you are concatenating by columns not by rows, right?

Great work, very helpfull for those who are starting.



Lathwal • Posted on Version 400 • 7 months ago • Options • Reply

^ 2 ▾



I will surely recommend this notebook for beginners !!

DATAI

DATAI Kernel Author

• Posted on Version 491 • 6 months ago • Options • Reply

^ 0 ▾

Thank you



Lathwal • Posted on Version 400 • 7 months ago • Options • Reply

^ 2 ▾



Great Notebook !!



Barthouze • Posted on Version 152 • 9 months ago • Options • Reply

^ 3 ▾



Awesome stuff here's a great resource to get started as well:

<https://jakevdp.github.io/PythonDataScienceHandbook/>

DATAI

DATAI Kernel Author

• Posted on Version 153 • 9 months ago • Options • Reply

^ 0 ▾

You are welcome



EDA • Posted on Version 240 • 8 months ago • Options • Reply

^ 1 ▾



Great Work! In the filtering section, 'query' is useful function to use as part of method chaining.

DATAI

DATAI Kernel Author

• Posted on Version 241 • 8 months ago • Options • Reply

^ 0 ▾

Thanks



Sean Fischer • Posted on Version 251 • 8 months ago • Options • Reply

^ 2 ▾



This is a really great summary of what you need to know. Nothing extra or confusing. Great Job!

DATAI

DATAI Kernel Author

• Posted on Version 251 • 8 months ago • Options • Reply

0



Thank you for your comment.



Dumitru Puscasu • Posted on Version 220 • 9 months ago • Options • Reply

1



Thank you for sharing. I have professional background in mathematics, SQL and Databases especially Oracle, but nevertheless I am beginner in Python and Data Science. I would like to say thanks to you for this great work definitely took a lot of time. It's helps me as a good references materials.

DATAI

DATAI Kernel Author

• Posted on Version 236 • 9 months ago • Options • Reply

0



You are welcome and thank you for your comment.



LK • Posted on Version 218 • 9 months ago • Options • Reply

1



Nice and tidy

DATAI

DATAI Kernel Author

• Posted on Version 236 • 9 months ago • Options • Reply

0



Thank you



wangs0622 • Posted on Version 218 • 9 months ago • Options • Reply

1



Sorry, I have a little question:

```
df.info()
# as you can see gender is object
# However if we use groupby, we can convert it categorical data.
# Because categorical data uses less memory, speed up operations like groupby
df["gender"] = df["gender"].astype("category")
df["treatment"] = df["treatment"].astype("category")
df.info()
```

As the result show, before we change the type to 'category', the memory usage is 208kb. However, After changing the type, the memory usage is 344kb. Why you say: 'Because categorical data uses less memory, speed up operations like groupby'?

Thank you~

DATAI Kernel Author • Posted on Version 236 • 9 months ago • Options • Reply

^ 1 ▼

DATAI



Well you got good point. Categorical data uses less memory when the memory matters not in small data like this and it uses less memory when machine learning techniques are used. Thank you for your attention and comment.



Luz aru • Posted on Version 217 • 9 months ago • Options • Reply

^ 1 ▼



Thanks Kaan for your excellent work, I really enjoy, very helpful.

DATAI

DATAI Kernel Author • Posted on Version 236 • 9 months ago • Options • Reply

^ 0 ▼



You are welcome



Dhruv Desai • Posted on Version 204 • 9 months ago • Options • Reply

^ 1 ▼



great job! Really helpful!

DATAI

DATAI Kernel Author • Posted on Version 212 • 9 months ago • Options • Reply

^ 0 ▼



Thanks



ZhangSiyuan • Posted on Version 200 • 9 months ago • Options • Reply

^ 1 ▼



It's a very good tutorial for beginners, thank for very much.

DATAI

DATAI Kernel Author • Posted on Version 201 • 9 months ago • Options • Reply

^ 0 ▼



You are welcome.



Sumanta Soren • Posted on Version 193 • 9 months ago • Options • Reply

^ 1 ▼



A comprehensive collection. It made by life bit easier.

DATAI

DATAI Kernel Author • Posted on Version 195 • 9 months ago • Options • Reply

^ 0 ▼



Thanks for your comment.



NikhilPandya • Posted on Version 191 • 9 months ago • Options • Reply

^ 1 ▼



Excellent kernel, really useful for beginners like me, serves as ready reference.

DATAI

DATAI Kernel Author

• Posted on Version 191 • 9 months ago • Options • Reply

^ 0 ▼



Thank you for your comment



Akash Srivastava • Posted on Version 185 • 9 months ago • Options • Reply

^ 1 ▼



thanks for this help, it really helped me to know these all.

DATAI

DATAI Kernel Author

• Posted on Version 185 • 9 months ago • Options • Reply

^ 0 ▼



You are welcome.



fanglf • Posted on Version 191 • 9 months ago • Options • Reply

^ 0 ▼



Nice work

Kaan Can wrote

You are welcome.



SwatiDadhich • Posted on Version 180 • 9 months ago • Options • Reply

^ 1 ▼



this is very useful, kudos :)

DATAI

DATAI Kernel Author

• Posted on Version 185 • 9 months ago • Options • Reply

^ 0 ▼



Thank you for your comment.



cxue34 • Posted on Version 179 • 9 months ago • Options • Reply

^ 1 ▼



Good work, thanks!

DATAI

DATAI Kernel Author

• Posted on Version 180 • 9 months ago • Options • Reply

^ 0 ▼



You are welcome



ndaedzo • Posted on Version 176 • 9 months ago • Options • Reply

^ 1 ▼



you just brought my interest back in data-science..thank you so much

DATAI

DATAI Kernel Author • Posted on Version 176 • 9 months ago • Options • Reply

••••

I am happy for it. You are welcome.

^ 0 ▼



Sheng • Posted on Version 176 • 9 months ago • Options • Reply

^ 1 ▼



This is really good. went through end to end. It touches every aspect of the data science from the start to the end without going through too much into each step. Thank you

DATAI

DATAI Kernel Author • Posted on Version 176 • 9 months ago • Options • Reply

••••

Thank you for your comment

^ 0 ▼



nirmalelumalai • Posted on Version 175 • 9 months ago • Options • Reply

^ 1 ▼



great work!

DATAI

DATAI Kernel Author • Posted on Version 176 • 9 months ago • Options • Reply

••••

Thanks

^ 0 ▼



Jianfeng • Posted on Version 172 • 9 months ago • Options • Reply

^ 1 ▼



really great work!! Appreciate it!

DATAI

DATAI Kernel Author • Posted on Version 172 • 9 months ago • Options • Reply

••••

You are welcome.

^ 0 ▼

George Roberts • Posted on Version 170 • 9 months ago • Options • Reply

^ 1 ▼



This is amazing, thanks so much for writing this! It really helps to have such a pedagogical analysis.



DATAI

DATAI Kernel Author

• Posted on Version 172 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome.



MrMRobot • Posted on Version 170 • 9 months ago • Options • Reply

^ 1 ▼

Good

DATAI

DATAI Kernel Author

• Posted on Version 172 • 9 months ago • Options • Reply

^ 0 ▼

Thanks



leitao • Posted on Version 167 • 9 months ago • Options • Reply

^ 1 ▼

Great, I just saw that you have updated the "Machine Learning" part. I press my gratitudes to your great job. I want to know when the "Statistical Thinking" part will be updated? Thank you!

DATAI

DATAI Kernel Author

• Posted on Version 168 • 9 months ago • Options • Reply

^ 3 ▼

Thank you for your comment. There will be deep learning tutorial with real application. I remove statistical thinking tutorial from my schedule for now.



leitao • Posted on Version 169 • 9 months ago • Options • Reply

^ 0 ▼

clear, thanks, I will enjoy it!



Dip Modi • Posted on Version 167 • 9 months ago • Options • Reply

^ 1 ▼

Thank You Very Much for putting great effort. It is very useful article :))

DATAI

DATAI Kernel Author

• Posted on Version 168 • 9 months ago • Options • Reply

^ 0 ▼

Thank you for your comment.



Antoine Le Merost • Posted on Version 166 • 9 months ago • Options • Reply

^ 1 ▼



Thanks, I needed that to understand the basic logic I'm back on the road now

Akash Ranjan • Posted on Version 164 • 9 months ago • Options • Reply

^ 1 ▼

Great tutorial for beginners. Thanks :)



Heitor Sasaki • Posted on Version 164 • 9 months ago • Options • Reply

^ 1 ▼

Thanks! Obrigado!

I have learnt a lot from this guide.



Sachin • Posted on Version 62 • 10 months ago • Options • Reply

^ 4 ▼



Great tutorial.

I would like to mention something, It is not a mistake in tutorial. I just want to confirm my doubt. Please correct me if I am wrong.

In Missing Data section when we do `data1 = data`. This does not create a copy. `data1` points to same memory location to as `data`. Any changes in `data1` or `data` is reflected in both.

So upon dropping NA in `data1` at `data1["Type 2"].dropna(inplace = True)` variable "data" also has no null values. which mean "fillna" call does nothing on data.

DATAI

DATAI Kernel Author • Posted on Version 63 • 10 months ago • Options • Reply

^ 0 ▼

You are totally right!!! In order to copy data frame, we can use `df_new = df.copy()` Thank you for your comment and question.



Amrish Kushwaha • Posted on Version 161 • 9 months ago • Options • Reply

^ 1 ▼



Great Stuff Buddy. I have learned and tried each of your code in my Notebook . Its great to learn basics of data science. Thanks a lot Kaan Can.

DATAI

DATAI Kernel Author • Posted on Version 162 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome. Thanks for your comment.



onder ay • Posted on Version 161 • 9 months ago • Options • Reply

^ 1 ▼



Helal sana kaardesim :D

DATAI Kernel Author • Posted on Version 162 • 9 months ago • Options • Reply

^ 0 ▼

DATAI

Thanks



刘彦超 • Posted on Version 161 • 9 months ago • Options • Reply

^ 1 ▼

This is very helpful to me and I hope I can create a model myself soon

DATAI

DATAI Kernel Author

• Posted on Version 162 • 9 months ago • Options • Reply

^ 0 ▼

Go for it.



Avi Garg • Posted on Version 161 • 9 months ago • Options • Reply

^ 1 ▼

nice

DATAI

DATAI Kernel Author

• Posted on Version 162 • 9 months ago • Options • Reply

^ 0 ▼

Thanks



surabhi • Posted on Version 157 • 9 months ago • Options • Reply

^ 1 ▼

Wow .. Thanks a lot, this is very concise and helpful

DATAI

DATAI Kernel Author

• Posted on Version 157 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome.



Vibhutha Kumara... • Posted on Version 157 • 9 months ago • Options • Reply

^ 1 ▼

Very Helpful.... (y)

DATAI

DATAI Kernel Author

• Posted on Version 157 • 9 months ago • Options • Reply

^ 0 ▼

Thank you for your comment.



Derrick M • Posted on Version 204 • 9 months ago • Options • Reply

^ 2 ▼



Thank you for sharing ..this definitely took a lot of time to put together

DATAI

DATAI Kernel Author

• Posted on Version 212 • 9 months ago • Options • Reply

^ 0 v

I thank you for your comment.

Raja Gopalakri... • Posted on Version 367 • 7 months ago • Options • Reply

^ 0 v

Can I fork this notebook to build my own kernel?

DATAI

DATAI Kernel Author

• Posted on Version 377 • 7 months ago • Options • Reply

^ 0 v

You can do what ever you want.



Nicholas Mucia • Posted on Version 153 • 9 months ago • Options • Reply

^ 1 v



Thank you, this was very helpful.

DATAI

DATAI Kernel Author

• Posted on Version 154 • 9 months ago • Options • Reply

^ 0 v

You are welcome.



Wibnu Dickson • Posted on Version 152 • 9 months ago • Options • Reply

^ 1 v



This helped me a lot , one of the best tutorial to have a head start in python .

DATAI

DATAI Kernel Author

• Posted on Version 153 • 9 months ago • Options • Reply

^ 0 v

Thank you for your comment.



Behcet Senturk • Posted on Version 143 • 9 months ago • Options • Reply

^ 1 v



Best tutorial for beginners like me :) .Thank you and do you have any advice for beginners like me?

DATAI

DATAI Kernel Author

• Posted on Version 144 • 9 months ago • Options • Reply

^ 0 v

Thank you for your comment. 1) Learn very well basics like numpy, pandas and seaborn. 2) Try to apply them on real life problem. 3) Create your own kernel, this makes you better understanding of what your kernel about 4) Stay in follow :)



Behcet Senturk • Posted on Version 148 • 9 months ago • Options • Reply

^ 0 ▼

Ok i will try. Thank you :)



Raja Gopalakrishnan • Posted on Version 392 • 7 months ago • Options • Reply

^ 0 ▼

Hi there, This is awesome for beginners like me. While I noticed how to retrieve a CSV file type from a location and reading it via Python, is there a way to query the CSV file via SQL within the same kernel? Can you mention an example, if it is possible?



kwanhooon • Posted on Version 143 • 9 months ago • Options • Reply

^ 1 ▼



Thx!

DATAI Kernel Author • Posted on Version 144 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome



Zach Eich • Posted on Version 143 • 9 months ago • Options • Reply

^ 1 ▼



Thank you, this is a pretty comprehensive intro.

DATAI Kernel Author • Posted on Version 144 • 9 months ago • Options • Reply

^ 0 ▼

Thank you for your comment.



DATAI Kernel Author • Posted on Version 32 • a year ago • Options • Reply

^ 4 ▼



Vote !!! Which one do you want as next tutorial.

DATAI Kernel Author • Posted on Version 32 • a year ago • Options • Reply

^ 9 ▼

Statistical Thinking

DATAI Kernel Author • Posted on Version 32 • a year ago • Options • Reply

^ 28 ▼

Machine Learning

DATAI [Kernel Author](#) • Posted on Version 32 • a year ago • Options • Reply ^ 6 ▾

Deep Learning

Hammad Zah... [Kernel Author](#) • Posted on Version 59 • 10 months ago • Options • Reply ^ 0 ▾

ML please

DATAI [Kernel Author](#) • Posted on Version 124 • 10 months ago • Options • Reply ^ 0 ▾

<https://www.kaggle.com/kanncaa1/machine-learning-tutorial-for-beginners/>

DATAI [Kernel Author](#) • Posted on Version 603 • 5 months ago • Options • Reply ^ 0 ▾

There is rare visualization tools: <https://www.kaggle.com/kanncaa1/rare-visualization-tools>

DATAI [Kernel Author](#) • Posted on Version 609 • 5 months ago • Options • Reply ^ 0 ▾

Statistical learning: <https://www.kaggle.com/kanncaa1/basic-statistic-tutorial-for-beginners>

Willy • Posted on Version 138 • 10 months ago • Options • Reply ^ 1 ▾

Congratulations Kaan! Thanks for your effort. I feel proud to see a young man from Turkey does one of the best tutorial. I would like to add that even if I am not a complete beginner. I sometimes struggle to understand some of your code. More explanations of the examples can make it perfect.

DATAI [Kernel Author](#) • Posted on Version 138 • 10 months ago • Options • Reply ^ 0 ▾

Thank you for your kind message. You can just ask where you have problem or where you think more explanation is need. Because, although I try to explain all of concepts and examples, sometimes I can miss some points.

el amraoui Sohayb • Posted on Version 138 • 10 months ago • Options • Reply ^ 1 ▾

a perfect place to start your data science journey; thanks a lot for the effort.

DATAI [Kernel Author](#) • Posted on Version 138 • 10 months ago • Options • Reply ^ 0 ▾

You are welcome.



Imran Hashmi • Posted on Version 186 • 9 months ago • Options • Reply

^ 2 ▼



That is a cracking tutorial, thanks so much!

DATAI

DATAI Kernel Author

• Posted on Version 186 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome



Johannesss • Posted on Version 133 • 10 months ago • Options • Reply

^ 1 ▼



Thank you

DATAI

DATAI Kernel Author

• Posted on Version 136 • 10 months ago • Options • Reply

^ 0 ▼

You are welcome



Diego Cavalca • Posted on Version 180 • 9 months ago • Options • Reply

^ 2 ▼



GREAT tutorial, Kaan! Thanks for sharing!!

It's an excellent resource for beginners in data science, like me.

It's ok for you if I translate it to Portuguese (pt-BR)? Sadly, there's a lack of good materials about DS (like it!) in our native language.

Obviously, I'll ATTACH your name.. :)

Looking foward for your reply.

DATAI

DATAI Kernel Author

• Posted on Version 180 • 9 months ago • Options • Reply

^ 6 ▼



It is public. You can make what ever you want. Go for it and send me the link:) Thank you for your comment



Zhu Yuhao • Posted on Version 129 • 10 months ago • Options • Reply

^ 1 ▼



Thanks for your contributions!

DATAI

DATAI Kernel Author

• Posted on Version 130 • 10 months ago • Options • Reply

^ 0 ▼

You are welcome



Shiva Manhar • Posted on Version 129 • 10 months ago • Options • Reply

^ 1 ▾



Thank you, this is useful for me. I think R and Python both are similar but SQL different.

DATAI

DATAI Kernel Author • Posted on Version 130 • 10 months ago • Options • Reply

^ 0 ▾

Although I do not know R, I think in some cases R is better and some other cases python is better.



GSD • Posted on Version 128 • 10 months ago • Options • Reply

^ 1 ▾



That was an exhaustive tutorial...It helped me a lot..Thank you ..

DATAI

DATAI Kernel Author • Posted on Version 130 • 10 months ago • Options • Reply

^ 0 ▾

Thank you for your comment.



JeffKing • Posted on Version 128 • 10 months ago • Options • Reply

^ 1 ▾



Appreciate the response. Deep Learning or Image Data Analysis next. Please! Thank you.

DATAI

DATAI Kernel Author • Posted on Version 130 • 10 months ago • Options • Reply

^ 0 ▾

There will be deep learning in the future but I do not really know when. I thank you



Haroon Ahmed • Posted on Version 127 • 10 months ago • Options • Reply

^ 1 ▾



Thanks this helps a lot !!

DATAI

DATAI Kernel Author • Posted on Version 127 • 10 months ago • Options • Reply

^ 0 ▾

You are welcome.



Kubilay B. • Posted on Version 127 • 10 months ago • Options • Reply

^ 1 ▾



Very concise, very clear, very Kaan :) I am proud of you, keep up the good work!

DATAI Kernel Author • Posted on Version 127 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

DATAI



Comments from people makes me happy, Comments from my followers makes my day better and encourage me But comments from my idol are unbelievable, incredible, fabulous and endless beautiful words :)



JeffKing • Posted on Version 127 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Thank you very much. This has been very useful and I believe it has motivated me to review the other tutorial (ML) and learn a lot again. Could you just do some examples on how to take care of missing values like ? and NaN. Also what should one do when there are too many 0 in the dataset? Thank you

DATAI



DATAI Kernel Author • Posted on Version 127 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

The first think is that it can be valuable to ask why there are zeros, do all zeros have a meaning or not. If the answer is no they do not have meaning, you can just drop them or you can assign them mean of data or add a constant value for each value of variable then take a log transformation. Thank you for your comment



Marius • Posted on Version 125 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Thanks, this is very useful and clear.

DATAI



DATAI Kernel Author • Posted on Version 127 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

You are welcome.



dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Thank you very much for tutorial , looking forward to deep learning tutorial !! Cheers. Sorry for repeated post it s due to connection error

DATAI



DATAI Kernel Author • Posted on Version 127 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Thank you for your comment. ML tutorial is published so you can look at it. There is more time for deep learning tutorial. Stay in follow.



ThomasDK • Posted on Version 125 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Thanks for the tutorial!

DATAI

DATAI Kernel Author • Posted on Version 125 • 10 months ago • Options • Reply

[^](#) **0** [▼](#)

Do not stop continue with part 2 machine learning tutorial Thank you for your comment.



Ahmet Tortumlu • Posted on Version 170 • 9 months ago • Options • Reply

[^](#) **2** [▼](#)

Great job Kaan diyeceğim ama ayıp olacak:) Eline sağlık, adını görünce gururlandım.

DATAI

DATAI Kernel Author • Posted on Version 172 • 9 months ago • Options • Reply

[^](#) **1** [▼](#)

Thank you for your comment. Yorumun için çok teşekkürler, beni çok mutlu etti.



jjn710 • Posted on Version 121 • 10 months ago • Options • Reply

[^](#) **1** [▼](#)

Very helpful:) Also looking forward to the next Statistical Thinking and ML, you explain in a very easy understanding way!

DATAI

DATAI Kernel Author • Posted on Version 124 • 10 months ago • Options • Reply

[^](#) **0** [▼](#)

Thanks. <https://www.kaggle.com/kanncaa1/machine-learning-tutorial-for-beginners/>



Tânia • Posted on Version 120 • 10 months ago • Options • Reply

[^](#) **1** [▼](#)

Great tutorial! Waiting for ML :)

DATAI

DATAI Kernel Author • Posted on Version 124 • 10 months ago • Options • Reply

[^](#) **0** [▼](#)

Thanks <https://www.kaggle.com/kanncaa1/machine-learning-tutorial-for-beginners/>



Barath Raja • Posted on Version 119 • 10 months ago • Options • Reply

[^](#) **1** [▼](#)

Thank you 😊 please add Statistical Thinking and ML soon

DATAI

DATAI Kernel Author • Posted on Version 119 • 10 months ago • Options • Reply

[^](#) **1** [▼](#)

Thank you for your comment.



Shivakumar Panu... • Posted on Version 118 • 10 months ago • Options • Reply

^ 1 ▾



excellent job!! Please include Machine Learning soon. Thanks :)

DATAI

DATAI Kernel Author

• Posted on Version 119 • 10 months ago • Options • Reply

^ 0 ▾

Last three day for ml tutorial :)



NeYo • Posted on Version 114 • 10 months ago • Options • Reply

^ 1 ▾



Very concise ,comprehensive and informative kernel. Thanks for making this tutorial.

DATAI

DATAI Kernel Author

• Posted on Version 114 • 10 months ago • Options • Reply

^ 0 ▾

Thank you for your comment



Hao • Posted on Version 161 • 9 months ago • Options • Reply

^ 2 ▾



WOW Thank you , great tutorial

DATAI

DATAI Kernel Author

• Posted on Version 162 • 9 months ago • Options • Reply

^ 0 ▾

You are welcome.



Zhongling Jiang • Posted on Version 113 • 10 months ago • Options • Reply

^ 1 ▾



It is very helpful. Thanks you!

DATAI

DATAI Kernel Author

• Posted on Version 113 • 10 months ago • Options • Reply

^ 0 ▾

You are welcome



Susannah Klaneč... • Posted on Version 159 • 9 months ago • Options • Reply

^ 2 ▾



Thanks!

DATAI Kernel Author

• Posted on Version 160 • 9 months ago • Options • Reply

^ 0 ▾

DATAI

You are welcome.



Mathias Meldgaa... • Posted on Version 109 • 10 months ago • Options • Reply



Excellent kernel. Good job!

DATAI

DATAI Kernel Author • Posted on Version 109 • 10 months ago • Options • Reply

••••

Thank you for your comment



Ashin George • Posted on Version 106 • 10 months ago • Options • Reply



Useful tutorial. Thanks :)

DATAI

DATAI Kernel Author • Posted on Version 109 • 10 months ago • Options • Reply

••••

You are welcome



Jagan Mohan Bat... • Posted on Version 106 • 10 months ago • Options • Reply



Awesome . Bookmarked. Good knowledge Repository. You spent a lot of time to share good information with all. Kudos to you.

DATAI

DATAI Kernel Author • Posted on Version 109 • 10 months ago • Options • Reply

••••

Thank you for your comment



Kalyan Lahkar • Posted on Version 104 • 10 months ago • Options • Reply



Very concise and informative. Thanks a ton.

DATAI

DATAI Kernel Author • Posted on Version 104 • 10 months ago • Options • Reply

••••

You are welcome.



Sharath Kothapalli • Posted on Version 104 • 10 months ago • Options • Reply

^ 1 ▼

Great kernel Kaan, Very Informative. It would be of great help if you through more light in Visual EDA. Explaining type of plotting used in specific Scenario, for example you have mentioned about the Scatter plot which is better when there is correlation between two variables. Thank you.

DATAI

DATAI Kernel Author

• Posted on Version 104 • 10 months ago • Options • Reply

^ 0 ▼

•••••

Thank you for your suggestion and comment.



Fabiano Bizarro • Posted on Version 104 • 10 months ago • Options • Reply

^ 1 ▼

Awesome, thanks

DATAI

DATAI Kernel Author

• Posted on Version 104 • 10 months ago • Options • Reply

^ 0 ▼

•••••

Thanks.



ledaiduong • Posted on Version 102 • 10 months ago • Options • Reply

^ 1 ▼

I am so happy when to learn the excellent tutorial like that, but i am expecting the next term of this tutorial with satistical thining and machine learning, please tell us when you can continue write it ?

DATAI

DATAI Kernel Author

• Posted on Version 104 • 10 months ago • Options • Reply

^ 0 ▼

•••••

I am now even exactly now writing ml tutorial. However, I will not continue with pokemon dataset. I hope it will be published at the end of this month. After ml tutorial, I will continue with deep learning. After all of these there will be statistical thinking. Thank you for your comment



Tony • Posted on Version 100 • 10 months ago • Options • Reply

^ 1 ▼

Thanks for such great kernel Kaan. But I couldn't understand pivot, stacking, unstacking, swaplevel. Is there any other resource for help?

DATAI

DATAI Kernel Author

• Posted on Version 101 • 10 months ago • Options • Reply

^ 1 ▼

•••••



Look at my other kernel. It is not tutorial but I used melting in swarm plot. I hope you can understand it with this example. <https://www.kaggle.com/kanncaa1/feature-selection-and-data-visualization> If you have trouble with melting or other things mentioned, watching video in youtube is best option that I usually make.



DanielGrzenda • Posted on Version 99 • 10 months ago • Options • Reply

^ 1 ▾



Nice! Thanks!

DATAI

DATAI Kernel Author

• Posted on Version 99 • 10 months ago • Options • Reply

^ 0 ▾

•••••

You are welcome.



Nad13 • Posted on Version 98 • 10 months ago • Options • Reply

^ 1 ▾



Great kernel, thanks!

DATAI

DATAI Kernel Author

• Posted on Version 99 • 10 months ago • Options • Reply

^ 0 ▾

•••••

Thank you for your comment.

MERKABAHNK



Juan R • Posted on Version 96 • 10 months ago • Options • Reply

^ 1 ▾

Hello, newbie question here, It's regarding the first Line Plot graph. Q - Does the Y axis correspond to the values in the Speed column? (I think yes) - also what does the X axis value correspond to (assume it's the number of different speeds) & it what order are they plotted? [Line Plot Graph](#)

DATAI

DATAI Kernel Author

• Posted on Version 100 • 10 months ago • Options • Reply

^ 0 ▾

•••••

Yes, it is Speed. If x axis is not assigned with special value or column, it becomes index. Therefore it is Speed vs index. Index is from zero to 800 as you can see in data.head() at first column. Actually it is not column, index is leftmost in data.head(). Thanks for your comment and question.



TommyYang • Posted on Version 94 • 10 months ago • Options • Reply

^ 1 ▾



Thanks, man. Really helpful and informative.

DATAI

DATAI Kernel Author

• Posted on Version 99 • 10 months ago • Options • Reply

^ 0 ▾

•••••

You are welcome.



Massimiliano Bot... • Posted on Version 92 • 10 months ago • Options • Reply

^ 1 ▾



Great introduction to Data Science with Python! +++++

DATAI**DATAI Kernel Author**

• Posted on Version 93 • 10 months ago • Options • Reply

[^](#) **0** [v](#)

Thank you for your comment.



Yunzhi Wu • Posted on Version 92 • 10 months ago • Options • Reply

[^](#) **1** [v](#)

Hi! Some stupid questions from a newbie: What is purpose of melt? set_index groups data in a more readable way, does it help ML also? What is difference between series and frame?

Thank you very much for your tutorial! Looking forwards to your ML part.

Best Regards, Yunzhi

DATAI**DATAI Kernel Author**

• Posted on Version 93 • 10 months ago • Options • Reply

[^](#) **0** [v](#)

It allows you to unpivot data that is not given readable way. Or in order to visualize data in seaborn you need to melt data. Look at my swarm plots in feature selection and visualization kernel. Series: one-dimensional labeled array capable of holding any data type DataFrame: 2-dimensional labeled data structure with columns of potentially different types. Like a sql table DataFrames takes Series as an input. Shortly, Series is a single column of a Data Frame. Thank you for your comment.



Yunzhi Wu • Posted on Version 99 • 10 months ago • Options • Reply

[^](#) **0** [v](#)

Thank you very much for your answering!



Joaquim Procopio • Posted on Version 91 • 10 months ago • Options • Reply

[^](#) **1** [v](#)

Nice.

DATAI**DATAI Kernel Author**

• Posted on Version 93 • 10 months ago • Options • Reply

[^](#) **0** [v](#)

Thanks



Chris Snipp • Posted on Version 90 • 10 months ago • Options • Reply

[^](#) **1** [v](#)

Great stuff! Thanks!

DATAI**DATAI Kernel Author**

• Posted on Version 93 • 10 months ago • Options • Reply

[^](#) **0** [v](#)

You are welcome.



SimBot • Posted on Version 89 • 10 months ago • Options • Reply

^ 1 ▼



Very useful tutorial! Thank you!

DATAI

DATAI Kernel Author

• Posted on Version 100 • 10 months ago • Options • Reply

^ 0 ▼

Thanks



Janio Alexander B... • Posted on Version 87 • 10 months ago • Options • Reply

^ 1 ▼



Awsome! Thanks for sharing! :)

DATAI

DATAI Kernel Author

• Posted on Version 87 • 10 months ago • Options • Reply

^ 0 ▼

Thanks :)



Ahmad M. • Posted on Version 86 • 10 months ago • Options • Reply

^ 1 ▼



Nice introductions for newbies...(not me) but newbies ^_^

DATAI

DATAI Kernel Author

• Posted on Version 86 • 10 months ago • Options • Reply

^ 0 ▼

Thanks :)



Avinash • Posted on Version 84 • 10 months ago • Options • Reply

^ 1 ▼



Thanks for the tutorial. It is very informative.

DATAI

DATAI Kernel Author

• Posted on Version 86 • 10 months ago • Options • Reply

^ 0 ▼

You are welcome.



Paul Ma • Posted on Version 135 • 10 months ago • Options • Reply

^ 2 ▼



Thanks for the work in putting this together.

DATAI

DATAI Kernel Author

• Posted on Version 136 • 10 months ago • Options • Reply

^ 0 ▼

Thank you for your comment



lalit • Posted on Version 84 • 10 months ago • Options • Reply

^ 1 ▼

I have one doubt regarding box plot in section In/Out[45]. What are black the circles above and below of min/max line ? Forgive me if I have missed something very obvious.

DATAI

DATAI Kernel Author

• Posted on Version 84 • 10 months ago • Options • Reply

^ 0 ▼

These are outliers that I mention at between 42-43. Or you can search with ctrl-f. Thanks for your question.



wyhao31 • Posted on Version 182 • 9 months ago • Options • Reply

^ 0 ▼

You mentioned "# There are no outliers" In In[45]. Also, I'm wondering why the black circles are outliers? How can the system detect it? Thanks.

DATAI

DATAI Kernel Author

• Posted on Version 186 • 9 months ago • Options • Reply

^ 0 ▼

I explain it in exploratory data analysis. It is shortly that outlier are smaller than $Q1 - 1.5(Q3-Q1)$ and bigger than $Q3 + 1.5(Q3-Q1)$ where value at 75% is Q3 and value at 25% is Q1



T.Meyer • Posted on Version 83 • 10 months ago • Options • Reply

^ 1 ▼

I am quite new to Python. This tutorial was very helpful to understand some basics - thank you! Will definitely look into your ML tutorial as well.

DATAI

DATAI Kernel Author

• Posted on Version 83 • 10 months ago • Options • Reply

^ 1 ▼

Thank you for your comment.



IshanPuranik • Posted on Version 82 • 10 months ago • Options • Reply

^ 1 ▼

thanks for the tutorial. Looking forward to the upcoming bits of it



DATAI Kernel Author

• Posted on Version 82 • 10 months ago • Options • Reply

^ 1 ▼

DATAI



There will be machine learning tutorial until the end of this month. I hope :)



[Deleted User] • Posted on Version 81 • 10 months ago • Options • Reply

^ 1 ▼



Thanks a lot , very informative !

DATAI

DATAI Kernel Author

• Posted on Version 82 • 10 months ago • Options • Reply



Thanks for your comment

^ 0 ▼



AlexTru • Posted on Version 81 • 10 months ago • Options • Reply

^ 1 ▼



This is awesome! Thank you!

DATAI

DATAI Kernel Author

• Posted on Version 82 • 10 months ago • Options • Reply



You are welcome.

^ 0 ▼



Zak Krumlinde • Posted on Version 79 • 10 months ago • Options • Reply

^ 1 ▼



This is great! I am an aspiring data scientist and this a great guide to get started.

DATAI

DATAI Kernel Author

• Posted on Version 82 • 10 months ago • Options • Reply



Thank you for your comment.

^ 0 ▼



Kevin Querino • Posted on Version 78 • 10 months ago • Options • Reply

^ 1 ▼



Thank you sir!

DATAI

DATAI Kernel Author

• Posted on Version 78 • 10 months ago • Options • Reply



Thank you Kevin.

^ 0 ▼



DENG • Posted on Version 78 • 10 months ago • Options • Reply

^ 1 ▼



It is too awesome! I have to say it is a very good guide for me.Thank you very so much.I am looking forward to your ML tutorial.

DATAI**DATAI Kernel Author**

• Posted on Version 78 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thank you for your comment. Keep in follow for ml tutorial !



AlexandruFlorescu • Posted on Version 77 • 10 months ago • Options • Reply

^ 1 ▼

Hey, perhaps you could also build a more in-depth plotting tutorial? :) I would be quite interested in that. Thanks for your time.

DATAI**DATAI Kernel Author**

• Posted on Version 77 • 10 months ago • Options • Reply

^ 0 ▼

••••

If you do not check my other kernels just look at them for plotting. If you already check them, I can not promise to make in-depth plotting tutorial because it is not in my calendar but if I have free time and update my calender, I consider your wish :) Thank you for your comment.



AlexandruFlor... • Posted on Version 78 • 10 months ago • Options • Reply

^ 0 ▼

I'm a newby, sorry :) I think the other kernels will suffice, plus I found [this](#) which seemed quite comprehensive. Thanks and have a nice evening :)



Salomon Tetelepta • Posted on Version 77 • 10 months ago • Options • Reply

^ 1 ▼

Thanks a lot!

DATAI**DATAI Kernel Author**

• Posted on Version 77 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thanks



Avi Kej • Posted on Version 76 • 10 months ago • Options • Reply

^ 1 ▼

nice work!

DATAI**DATAI Kernel Author**

• Posted on Version 77 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thanks



Reeram Reddy K... • Posted on Version 75 • 10 months ago • Options • Reply

^ 1 ▾



This is awesome. Can you please add codes for Statistical thinking and Machine Learning also. Really looking forward for it

DATAI

DATAI Kernel Author • Posted on Version 75 • 10 months ago • Options • Reply

^ 0 ▾



Guess what :) Now I am working on ML tutorial :) And thank you for your supportive comment.



Reeram Redd... • Posted on Version 75 • 10 months ago • Options • Reply

^ 0 ▾



Thanks a lot. This kernel is really very helpful. If possible can you please add content on Deep Learning and other stuff.

Thanks a lot again !!!!!!



Evgeny Vlasov • Posted on Version 73 • 10 months ago • Options • Reply

^ 1 ▾



Perfect guide, man! Thank you. It inspires me for future learning with data sciende and ML :)

DATAI

DATAI Kernel Author • Posted on Version 74 • 10 months ago • Options • Reply

^ 0 ▾



You are welcome. Thank you for your comment.



Pavel • Posted on Version 73 • 10 months ago • Options • Reply

^ 1 ▾



Thank

DATAI

DATAI Kernel Author • Posted on Version 74 • 10 months ago • Options • Reply

^ 0 ▾



You are welcome.



ChunfeiXu • Posted on Version 70 • 10 months ago • Options • Reply

^ 1 ▾



This tutorial is very useful,thanks!

DATAI

DATAI Kernel Author • Posted on Version 72 • 10 months ago • Options • Reply

^ 0 ▾



Thank you for your comment.



Hector Huaranca • Posted on Version 70 • 10 months ago • Options • Reply

^ 1 ▼



Thank you Kaan for your tutorial, was very useful for me.

DATAI

DATAI Kernel Author

• Posted on Version 72 • 10 months ago • Options • Reply

^ 0 ▼

••••

You are welcome.



Chen Xinyan • Posted on Version 69 • 10 months ago • Options • Reply

^ 1 ▼



Thank you so much for the tutorial!

DATAI

DATAI Kernel Author

• Posted on Version 70 • 10 months ago • Options • Reply

^ 0 ▼

••••

You are welcome.



Chen Xinyan • Posted on Version 69 • 10 months ago • Options • Reply

^ 1 ▼



Thank you so much for the tutorial!

DATAI

DATAI Kernel Author

• Posted on Version 74 • 10 months ago • Options • Reply

^ 0 ▼

••••

You are welcome.



[Deleted User] • Posted on Version 67 • 10 months ago • Options • Reply

^ 1 ▼



Nice and helpful tutorial..

DATAI

DATAI Kernel Author

• Posted on Version 68 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thank you for your comment.



Karim Ali • Posted on Version 65 • 10 months ago • Options • Reply

^ 1 ▼



It was very well structured. Thank you for this tutorial!

DATAI

DATAI Kernel Author • Posted on Version 67 • 10 months ago • Options • Reply

 0 

You are welcome.



Divakar Chirrared... • Posted on Version 63 • 10 months ago • Options • Reply

 1 

Very nicely articulated. Thanks for sharing this tutorial.

DATAI

DATAI Kernel Author • Posted on Version 63 • 10 months ago • Options • Reply

 0 

Thanks for your good comment.



Bazinga • Posted on Version 61 • 10 months ago • Options • Reply

 1 

Thank you so much for the tutorial, extremely helpful.

DATAI

DATAI Kernel Author • Posted on Version 62 • 10 months ago • Options • Reply

 0 

You are welcome.



Bazinga • Posted on Version 61 • 10 months ago • Options • Reply

 1 

Thank you so much for the tutorial, extremely helpful.

DATAI

DATAI Kernel Author • Posted on Version 62 • 10 months ago • Options • Reply

 0 

You are welcome



Hammad Zahid Ali • Posted on Version 59 • 10 months ago • Options • Reply

 1 

notable :) great work

DATAI

DATAI Kernel Author • Posted on Version 59 • 10 months ago • Options • Reply

 0 

Thanks



陈铭嘉(Peter Chen) • Posted on Version 59 • 10 months ago • Options • Reply

^ 1 ▼

Hello. This is amazingly helpful.I wonder if it's ok to translate it into chinese.I want to let more people see it in china.
And your name will be attached.Looking forward for you reply. Thank you.

DATAI

DATAI Kernel Author

• Posted on Version 59 • 10 months ago • Options • Reply

^ 1 ▼

••••

If you translate it into Chinese and achieve more people, I will be very happy. Go for it and send me link please. Thank you for your offer and comment.



Masha • Posted on Version 108 • 10 months ago • Options • Reply

^ 2 ▼



That's a very comprehensive yet concise tutorial! Thanks a lot!

DATAI

DATAI Kernel Author

• Posted on Version 109 • 10 months ago • Options • Reply

^ 0 ▼

••••

You are welcome



PoojaW • Posted on Version 50 • 10 months ago • Options • Reply

^ 1 ▼



Beautifully explained! Thanks friend! Looking forward to your Machine Learning post.

DATAI

DATAI Kernel Author

• Posted on Version 53 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thank you. I am working on ml tutorial :)



Mohamed Shawk... • Posted on Version 49 • 10 months ago • Options • Reply

^ 1 ▼



Great work. thanks for the tutorial.

DATAI

DATAI Kernel Author

• Posted on Version 49 • 10 months ago • Options • Reply

^ 0 ▼

••••

You are welcome



Bharath • Posted on Version 48 • 10 months ago • Options • Reply

^ 1 ▼



Really nice intuitive explanation.

DATAI Kernel Author • Posted on Version 48 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

DATAI



Thank you.



Barath Raja • Posted on Version 47 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Thank you.

DATAI

DATAI Kernel Author

• Posted on Version 47 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

You are welcome



Cenk KIRAN • Posted on Version 44 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Çok güzel bir toplama olmuş eline sağlık!

Ufak bir katkı, Jupyter ile çalışacaklar için başa; %matplotlib inline eklemeleri faydalı olacaktır.

DATAI

DATAI Kernel Author

• Posted on Version 44 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

He writes: It has been a great summarizing! A small contribution, adding "%matplotlib inline" at the beginning will be useful for people who works with Jupyter. Thank you for your comment.

Yorumun icin teşekkür ederim. Bu tarz yorumlar hem tesvik edici oluyor hemde moral veriyor.



asthascientist • Posted on Version 43 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

This is very helpful - I have forked it!

DATAI

DATAI Kernel Author

• Posted on Version 43 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Thank you.



richarde • Posted on Version 43 • 10 months ago • Options • Reply

[^](#) [1](#) [▼](#)

very clear and concise tutorial

DATAI

DATAI Kernel Author

• Posted on Version 43 • 10 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Thank you.



Eero Lehtonen • Posted on Version 41 • 10 months ago • Options • Reply

^ 1 ▼



Thanks man! This was really good stuff. I vote for machine learning.

DATAI

DATAI Kernel Author

• Posted on Version 41 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thanks. I hope there will be machine learning tutorial in this month.



刘彦超 • Posted on Version 41 • 10 months ago • Options • Reply

^ 1 ▼



This is helpful for me, Thank you.

DATAI

DATAI Kernel Author

• Posted on Version 41 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thank you for your comment.



PolleTuurimeRon... • Posted on Version 40 • 10 months ago • Options • Reply

^ 1 ▼



This is amazingly helpful. Great work. I vote for machine learning.

DATAI

DATAI Kernel Author

• Posted on Version 40 • 10 months ago • Options • Reply

^ 0 ▼

••••

Thanks for your comment.



Aman Batra • Posted on Version 37 • a year ago • Options • Reply

^ 1 ▼



Thanks for this, Loving it

DATAI

DATAI Kernel Author

• Posted on Version 37 • a year ago • Options • Reply

^ 0 ▼

••••

You are welcome



••

Koen Wolters • Posted on Version 30 • a year ago • Options • Reply

^ 1 ▼



Thanks for this great tutorial

DATAI

DATAI Kernel Author

• Posted on Version 30 • a year ago • Options • Reply

 0 

Thanks for your comment.



Devaraju Vinoda • Posted on Version 29 • a year ago • Options • Reply

 1 

Its like benchmark of concepts for beginners in data-science, it really helped me so as to what to study and what not to. Thank you for this awesome tutorial.

DATAI

DATAI Kernel Author

• Posted on Version 29 • a year ago • Options • Reply

 1 

Stay in follow! This is just the beginning. Thank you for your comment.



Ashwin Kandel • Posted on Version 28 • a year ago • Options • Reply

 1 

Thank you for your time and effort to create this. Really helpful for novice like myself.

DATAI

DATAI Kernel Author

• Posted on Version 28 • a year ago • Options • Reply

 0 

I hope you are going to be Contributor and learn basics of data analysis very well. Because there are going to be more advance topics like machine learning, even deep learning. Stay in follow :)



Andrea Seveso • Posted on Version 27 • a year ago • Options • Reply

 1 

Learning using examples is the best way to learn fast. Thanks a lot for your notebook, really useful.

DATAI

DATAI Kernel Author

• Posted on Version 28 • a year ago • Options • Reply

 0 

Yes, I agree with you. Thanks for your comment.



ZhQing • Posted on Version 82 • 10 months ago • Options • Reply

 2 

Wonderful tutorial! Thank you so much. And I can't wait to see your ML tutorial, when will you publish it?

DATAI

DATAI Kernel Author

• Posted on Version 82 • 10 months ago • Options • Reply

 0 

Thank you for your comment. There are a lot of thing to put into ml tutorial. I am trying to organize schedule of it because it should not be too hard for beginners and also be informative enough.

Therefore, I hope until the end of this month I will publish ml tutorial kernel.



BHolland • Posted on Version 26 • a year ago • Options • Reply

^ 1 ▼



This looks fantastic! Great work!

DATAI

DATAI Kernel Author

• Posted on Version 28 • a year ago • Options • Reply

^ 0 ▼

Thanks !



ericthebren • Posted on Version 26 • a year ago • Options • Reply

^ 1 ▼



Thank you for posting this! Great intro.

DATAI

DATAI Kernel Author

• Posted on Version 28 • a year ago • Options • Reply

^ 0 ▼

Thank you for your comment.



Shun nakamura • Posted on Version 24 • a year ago • Options • Reply

^ 1 ▼



Thank you for your great work! It's very useful for me!

DATAI

DATAI Kernel Author

• Posted on Version 24 • a year ago • Options • Reply

^ 0 ▼

I thank you for your comment



nihal88 • Posted on Version 24 • a year ago • Options • Reply

^ 1 ▼



Great guide for beginner like me , Thanks

DATAI

DATAI Kernel Author

• Posted on Version 24 • a year ago • Options • Reply

^ 0 ▼

You are welcome.



Kheirallah Samah... • Posted on Version 24 • a year ago • Options • Reply

^ 1 ▼



Very useful indeed...

thank for sharing

DATAI

DATAI Kernel Author

• Posted on Version 24 • a year ago • Options • Reply



Thank you for your comment.

^ 1 ▼



Ben Gordon • Posted on Version 20 • a year ago • Options • Reply

^ 1 ▼



Very useful. Thank you for putting this together.

DATAI

DATAI Kernel Author

• Posted on Version 20 • a year ago • Options • Reply



You are welcome.

^ 1 ▼



J G Keerthi Chath... • Posted on Version 19 • a year ago • Options • Reply

^ 1 ▼



great job...tnx

DATAI

DATAI Kernel Author

• Posted on Version 19 • a year ago • Options • Reply



Thank you

^ 1 ▼



Yasar Kucukefe • Posted on Version 18 • a year ago • Options • Reply

^ 1 ▼



Thank you for this great tutorial.

DATAI

DATAI Kernel Author

• Posted on Version 18 • a year ago • Options • Reply



You are welcome. Thank you for your comment

^ 1 ▼



AlexandruFlorescu • Posted on Version 71 • 10 months ago • Options • Reply

^ 2 ▼



hey, great tutorial, I learned a lot. Just wanted to mention it's tuple not table :D Also looking forward to the ML tutorial :) Thanks for your time

DATAI

DATAI Kernel Author

• Posted on Version 72 • 10 months ago • Options • Reply

^ 3 ▼

Thank you for your comment. I have finished the first part of the ML tutorial. There will be 3 part in total. When all of them be ready, I will publish :)



ROkabe • Posted on Version 57 • 10 months ago • Options • Reply

^ 2 ▼



Excellent for beginners like me. Many thanks. I vote Machine Learning for pt 2 :)

DATAI

DATAI Kernel Author • Posted on Version 57 • 10 months ago • Options • Reply

^ 0 ▼

Stay in follow. Machine learning will be published in this month



Gowtham Kumar • Posted on Version 56 • 10 months ago • Options • Reply

^ 2 ▼



Nice Tutorial!. well explained.

DATAI

DATAI Kernel Author • Posted on Version 57 • 10 months ago • Options • Reply

^ 0 ▼

Thank you for your comment



Jongjin Sohn • Posted on Version 56 • 10 months ago • Options • Reply

^ 2 ▼



Great tutorial! Many thanks!

DATAI

DATAI Kernel Author • Posted on Version 59 • 10 months ago • Options • Reply

^ 0 ▼

Thank you for your comment.



Cindy • Posted on Version 45 • 10 months ago • Options • Reply

^ 2 ▼



Thank you for sharing :)

DATAI

DATAI Kernel Author • Posted on Version 45 • 10 months ago • Options • Reply

^ 0 ▼

Thank you for comment :)

Uche Mbaka • Posted on Version 26 • a year ago • Options • Reply

^ 2 ▼



This is Awesome. Thanks a lot. On the part below, I think you meant Tuple

***tuple: sequence of immutable python objects. cant modify values tuple uses parenthesis like tuple = (1,2,3)
unpack tuple into several variables like a,b,c = tuple *

DATAI

DATAI Kernel Author

• Posted on Version 28 • a year ago • Options • Reply

^ 0 v
•••••

Thanks. Yes, I meant Tuple :)



Damir Shakenov • Posted on Version 16 • a year ago • Options • Reply

^ 2 v


Nice work, mate!

You've counted incorrect median in 3 Part (EDA) between 42th and 43th.

```
list = [1,4,5,6,8,9,11,12,14,15,16,17]'  
median = np.median(list) # 10.0
```

DATAI

DATAI Kernel Author

• Posted on Version 18 • a year ago • Options • Reply

^ 1 v
•••••

Thank you for your comment and correction. You are right and I correct it while adding 13 between 12 and 14. list = [1,4,5,6,8,9,11,12,13,14,15,16,17]' median = np.median(list) # 11.0



Yossi Levy • Posted on Version 73 • 10 months ago • Options • Reply

^ -3 v

You must be kidding. This tutorial will not teach anyone how to be a data scientist. A data technician at most.
Maybe.



dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 v


dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 v


dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 v


dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 v


dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 v



dataartist99 • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 ▼



Marius • Posted on Version 125 • 10 months ago • Options • Reply

^ 0 ▼

Thanks, this is very useful and clear.



Haroon Ahmed • Posted on Version 127 • 10 months ago • Options • Reply

^ 0 ▼



seapea • Posted on Version 132 • 10 months ago • Options • Reply

^ 0 ▼

Is this how kernels are supposed to be done? It's extremely long and doesn't really make any concise points of any sort. It just seems like a notebook that was used to for some EDA.

DATAI

DATAI Kernel Author • Posted on Version 132 • 10 months ago • Options • Reply

^ 0 ▼

"Is this how kernels are supposed to be done?" No, it is not. "It's extremely long and doesn't really make any concise points of any sort" Yes, it is because it is for beginners. It is not only EDA but it also python tutorial. The key point is "for beginners" "It just seems like a notebook that was used to for some EDA". Yes, it is Thank you for your comments.



Manikanta Sreek... • Posted on Version 226 • 9 months ago • Options • Reply

^ 0 ▼

Thanks for clear explanation

DATAI

DATAI Kernel Author • Posted on Version 236 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome.



Sahar Sela • Posted on Version 233 • 9 months ago • Options • Reply

^ 0 ▼

A very nice job with this tutorial! Thank you.

DATAI

DATAI Kernel Author • Posted on Version 236 • 9 months ago • Options • Reply

^ 0 ▼

You are welcome.



Qing Dai • Posted on Version 234 • 9 months ago • Options • Reply

^ 0 ∨

nice very nice

DATAI

DATAI Kernel Author

• Posted on Version 236 • 9 months ago • Options • Reply

^ 0 ∨



Thanks



lakshadvani • Posted on Version 235 • 9 months ago • Options • Reply

^ 0 ∨

This is a really good tutorial, will you be creating one with an in-depth math overview in the future?

DATAI

DATAI Kernel Author

• Posted on Version 236 • 8 months ago • Options • Reply

^ 0 ∨



I think the answer is no. There will be detailed deep learning tutorial in the feature.

DATAI

DATAI Kernel Author

• Posted on Version 236 • 8 months ago • Options • Reply

^ 0 ∨



I write wrong "feature" that is future :)



JimmyChen • Posted on Version 235 • 9 months ago • Options • Reply

^ 0 ∨

Thanks for making the guidebook! Very useful!

DATAI

DATAI Kernel Author

• Posted on Version 236 • 8 months ago • Options • Reply

^ 0 ∨



Thanks for your comment.



Jason • Posted on Version 248 • 8 months ago • Options • Reply

^ 0 ∨

Thank you very much

DATAI

DATAI Kernel Author

• Posted on Version 248 • 8 months ago • Options • Reply

^ 0 ∨



You are welcome.

Zijan Wu • Posted on Version 254 • 8 months ago • Options • Reply

^ 0 ∨



Thanks for your work. It's very useful. But maybe you can add `plot.show()` before show a figure. And in the part of loop, you mean archive or achieve?

DATAI**DATAI Kernel Author**

• Posted on Version 258 • 8 months ago • Options • Reply



Thanks for your comment. It is achieve.



Farid Muradov • Posted on Version 261 • 8 months ago • Options • Reply

tshk abi! cok faydası oldu bana.

DATAI**DATAI Kernel Author**

• Posted on Version 261 • 8 months ago • Options • Reply



Çok sevindim. You are welcome.



Movses Elbakian • Posted on Version 262 • 8 months ago • Options • Reply

Great Notebook!

DATAI**DATAI Kernel Author**

• Posted on Version 262 • 8 months ago • Options • Reply



Thanks



Vanish007 • Posted on Version 262 • 8 months ago • Options • Reply

Thank you for the tutorial, very informative! Is there a way to add a trendline to the scatter plot of "Attack" and "Defense"? Appreciate it!

DATAI**DATAI Kernel Author**

• Posted on Version 262 • 8 months ago • Options • Reply



As I know there is no library for trendline but you can add it while finding points. However is it correct idea to add trendline to attack defense scatter plot?



Vanish007 • Posted on Version 273 • 8 months ago • Options • Reply

Granted trend lines are more likely to be added when correlations are closer to 1, but would it be wrong to add one here? Would it not be just another model to agree with the previous pearson's correlation? Please correct me if I'm wrong =)

DATAI

DATAI Kernel Author • Posted on Version 285 • 8 months ago • Options • Reply

^ 2 ▼

Yes, it would. You are right :)



HenriqueOliveira • Posted on Version 262 • 8 months ago • Options • Reply

^ 0 ▼

Great work!

DATAI

DATAI Kernel Author • Posted on Version 262 • 8 months ago • Options • Reply

^ 0 ▼

Thank you



Jared • Posted on Version 265 • 8 months ago • Options • Reply

^ 0 ▼

Thanks! Nice tutorial for beginners.

DATAI

DATAI Kernel Author • Posted on Version 267 • 8 months ago • Options • Reply

^ 0 ▼

You are welcome.



Rajeev Warrier • Posted on Version 270 • 8 months ago • Options • Reply

^ 0 ▼

Thank you so much for this. Very well put together and incredibly helpful!

DATAI

DATAI Kernel Author • Posted on Version 285 • 8 months ago • Options • Reply

^ 0 ▼

You are welcome.



prashant • Posted on Version 271 • 8 months ago • Options • Reply

^ 0 ▼

thanks.... i was looking for this.

DATAI

DATAI Kernel Author • Posted on Version 285 • 8 months ago • Options • Reply

^ 0 ▼

I am happy for you found it.



Mehmet Fırat KU... • Posted on Version 2/3 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

This tutorial is great! thanks for the informations

DATAI

DATAI Kernel Author

• Posted on Version 285 • 8 months ago • Options • Reply

[^](#) [1](#) [▼](#)

Thank you for your comment.



Bo-Qiang Lu • Posted on Version 275 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Good

DATAI

DATAI Kernel Author

• Posted on Version 285 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Thanks



Nurrizky I • Posted on Version 277 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Great works!

DATAI

DATAI Kernel Author

• Posted on Version 285 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Thanks



Eduardo Martín Iz... • Posted on Version 281 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Great job dude but you are repeating some concepts

DATAI

DATAI Kernel Author

• Posted on Version 285 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

There can be repeating you are right. But it is good for beginners :)



Yaniv Bordeynik • Posted on Version 285 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Great Job!

DATAI

DATAI Kernel Author

• Posted on Version 285 • 8 months ago • Options • Reply

[^](#) [0](#) [▼](#)

Thank you



SmrutiranjanTripathi · Posted on Version 286 · 8 months ago · Options · Reply

^ 0 ▼

The best thing ever i came across about data science .Thanks a lot.

DATAI

DATAI Kernel Author · Posted on Version 293 · 8 months ago · Options · Reply



Thank you for your comment



Nirantak Raghav · Posted on Version 290 · 8 months ago · Options · Reply

^ 0 ▼

Good summary, useful for a quick reference

DATAI

DATAI Kernel Author · Posted on Version 293 · 8 months ago · Options · Reply



Thank you



Emmanuel Rosa · Posted on Version 293 · 8 months ago · Options · Reply

^ 0 ▼

Great work! Extremely helpful!

DATAI

DATAI Kernel Author · Posted on Version 293 · 8 months ago · Options · Reply



thanks



RudyMizrahi · Posted on Version 293 · 8 months ago · Options · Reply

^ 0 ▼

This is great from the flow to the examples!

DATAI

DATAI Kernel Author · Posted on Version 299 · 8 months ago · Options · Reply



Thanks



Danan Dio · Posted on Version 293 · 8 months ago · Options · Reply

^ 0 ▼

great man it's like head first for data science, keep it up

DATAI

DATAI Kernel Author

• Posted on Version 299 • 8 months ago • Options • Reply

^ 0 ∨

Thank you



GeekYoung • Posted on Version 294 • 8 months ago • Options • Reply

^ 0 ∨

Great! thank you very much!

DATAI

DATAI Kernel Author

• Posted on Version 302 • 8 months ago • Options • Reply

^ 0 ∨

You are welcome



GeekYoung • Posted on Version 294 • 8 months ago • Options • Reply

^ 0 ∨



GeekYoung • Posted on Version 294 • 8 months ago • Options • Reply

^ 0 ∨



Fatih • Posted on Version 296 • 8 months ago • Options • Reply

^ 0 ∨

Dostum ellerine sağlık...

DATAI

DATAI Kernel Author

• Posted on Version 299 • 8 months ago • Options • Reply

^ 0 ∨

Teşekkürler, faydalı olduysa ne mutlu bana.



VARUN R • Posted on Version 299 • 8 months ago • Options • Reply

^ 0 ∨

Thank you.....your tutorials helped me a lot.....!!!

DATAI

DATAI Kernel Author

• Posted on Version 302 • 8 months ago • Options • Reply

^ 0 ∨

You are welcome

Sangho • Posted on Version 301 • 8 months ago • Options • Reply

^ 0 ∨



Good work. Its' very helpful.

DATAI

DATAI Kernel Author

• Posted on Version 302 • 8 months ago • Options • Reply

^ 0 v

Thank you



Vamshi Krishna • Posted on Version 304 • 8 months ago • Options • Reply

^ 0 v

Very informative!

DATAI

DATAI Kernel Author

• Posted on Version 310 • 8 months ago • Options • Reply

^ 0 v

thanks



hknmtt • Posted on Version 311 • 8 months ago • Options • Reply

^ 0 v

Thank you very much for this very reader-friendly tutorial!

By the way, I found several typos that you can fix ;)

- 1) it should be 'tuple' rather than 'tuble'
- 2) the upper quartile in your example should be 14

DATAI

DATAI Kernel Author

• Posted on Version 313 • 8 months ago • Options • Reply

^ 0 v

As long as google understand me, I do not pay attention the spelling of words. Therefore, it can be confused sometimes tuple or table. But you are right it is tuple :) According to pd.describe() method it should be 14 but it also can be 14.5. Thank you for your comment.



御魂笑光辉 • Posted on Version 312 • 8 months ago • Options • Reply

^ 0 v

Thanks for your work, let me study the data science better.

DATAI

DATAI Kernel Author

• Posted on Version 318 • 7 months ago • Options • Reply

^ 0 v

You are welcome.



TopHighland • Posted on Version 316 • 8 months ago • Options • Reply

^ 0 v

It is really great for beginner

DATAI

DATAI Kernel Author

• Posted on Version 318 • 7 months ago • Options • Reply



Thank you

^ 0 ∨



Satya Thirumani • Posted on Version 317 • 8 months ago • Options • Reply

^ 0 ∨

Thank you so much for the tutorial...

DATAI

DATAI Kernel Author

• Posted on Version 318 • 7 months ago • Options • Reply



You are welcome.

^ 0 ∨



hariprasaath • Posted on Version 339 • 7 months ago • Options • Reply

^ 0 ∨

Thanks man. It's really worth for beginners.

DATAI

DATAI Kernel Author

• Posted on Version 341 • 7 months ago • Options • Reply



Thanks for your comment.

^ 0 ∨



nihal88 • Posted on Version 339 • 7 months ago • Options • Reply

^ 0 ∨

Very nice tutorial for a begginner like me . Thanks

DATAI

DATAI Kernel Author

• Posted on Version 341 • 7 months ago • Options • Reply



You are welcome.

^ 0 ∨



EdisonHuang • Posted on Version 343 • 7 months ago • Options • Reply

^ 0 ∨

This is so helpful!!! Thank you for sharing!

DATAI

DATAI Kernel Author

• Posted on Version 376 • 7 months ago • Options • Reply



You are welcome.

^ 0 ∨



walk2campus • Posted on Version 343 • 7 months ago • Options • Reply

^ 0 ▾

Thanks so much for your sharing. Great appreciate!

DATAI

DATAI Kernel Author

• Posted on Version 376 • 7 months ago • Options • Reply

^ 0 ▾

••••• Thank you for your comment.



Satyros • Posted on Version 347 • 7 months ago • Options • Reply

^ 0 ▾

Excellent! Thanks

DATAI

DATAI Kernel Author

• Posted on Version 376 • 7 months ago • Options • Reply

^ 0 ▾

••••• You are welcome.



Taha Tekdoğan • Posted on Version 352 • 7 months ago • Options • Reply

^ 0 ▾

Such a nice tutorial with good explanations! Thank you for sharing!

DATAI

DATAI Kernel Author

• Posted on Version 376 • 7 months ago • Options • Reply

^ 0 ▾

••••• Thanks :)



Manish • Posted on Version 362 • 7 months ago • Options • Reply

^ 0 ▾

This is simply awesome. I am new to data science and this really helps me to understand the what is data science and what you need to learn to become a good data scientist.

DATAI

DATAI Kernel Author

• Posted on Version 376 • 7 months ago • Options • Reply

^ 0 ▾

••••• Thank you for your awesome comment.



Jocelyn D'Souza • Posted on Version 363 • 7 months ago • Options • Reply

^ 0 ▾

Thanks for sharing!

DATAI

DATAI Kernel Author • Posted on Version 376 • 7 months ago • Options • Reply

^ 0 ▼

I thank you.



Muthu • Posted on Version 364 • 7 months ago • Options • Reply

^ 0 ▼

Really helped a beginner like me. Thanks

DATAI

DATAI Kernel Author • Posted on Version 376 • 7 months ago • Options • Reply

^ 0 ▼

You are welcome



Gui Gessele • Posted on Version 377 • 7 months ago • Options • Reply

^ 0 ▼

Great job :D Thanks for the help

DATAI

DATAI Kernel Author • Posted on Version 377 • 7 months ago • Options • Reply

^ 0 ▼

You are welcome.



Gnaneshwar • Posted on Version 389 • 7 months ago • Options • Reply

^ 0 ▼

Excellent kernel, really useful for beginners like me

DATAI

DATAI Kernel Author • Posted on Version 395 • 7 months ago • Options • Reply

^ 0 ▼

Thanks



MekalaGanesh • Posted on Version 392 • 7 months ago • Options • Reply

^ 0 ▼

it's very good for understanding i'm beginner please share any material for learning thank you

DATAI

DATAI Kernel Author • Posted on Version 395 • 7 months ago • Options • Reply

^ 0 ▼

You are welcome. Check my other courses * <https://www.kaggle.com/kanncaa1/deep-learning-tutorial-for-beginners> * <https://www.kaggle.com/kanncaa1/seaborn-tutorial-for-beginners> * <https://www.kaggle.com/kanncaa1/feature-selection-and-data-visualization>



Prit Ulmas • Posted on Version 411 • 6 months ago • Options • Reply

^ 0 ▾

Thanks a lot, such a great resource!

DATAI

DATAI Kernel Author • Posted on Version 425 • 6 months ago • Options • Reply



You are welcome

^ 0 ▾

• Posted on Version 412 • 6 months ago • Options • Reply

^ 0 ▾

DATAI

DATAI Kernel Author • Posted on Version 425 • 6 months ago • Options • Reply



You are welcome

^ 0 ▾

ryang lee • Posted on Version 420 • 6 months ago • Options • Reply

^ 0 ▾

Nice tutorial! thank you

DATAI

DATAI Kernel Author • Posted on Version 425 • 6 months ago • Options • Reply



Thanks

^ 0 ▾

Lokman • Posted on Version 424 • 6 months ago • Options • Reply

^ 0 ▾

Thank you very much (from a beginner)

DATAI

DATAI Kernel Author • Posted on Version 425 • 6 months ago • Options • Reply



Thank you for your comment

^ 1 ▾

Saurabh Sharma • Posted on Version 425 • 6 months ago • Options • Reply

^ 0 ▾

Thanks for the writeup. A small question though, I do not have any experience with Python/R. Which one should be preferred by a beginner/ Or, what should be the criteria for choosing?

DATAI Kernel Author • Posted on Version 431 • 6 months ago • Options • Reply

^ 1 ▼

DATAI



Both have advantages and disadvantages. As a first choice you learn both of them. Secondly, if you choose only one of them, choose python that has more source to learn.



Kalaba Chibale • Posted on Version 429 • 6 months ago • Options • Reply

^ 0 ▼

Very nice :)

DATAI



DATAI Kernel Author • Posted on Version 431 • 6 months ago • Options • Reply

^ 0 ▼

Thanks



Brondong • Posted on Version 430 • 6 months ago • Options • Reply

^ 0 ▼

Awesome job !

DATAI



DATAI Kernel Author • Posted on Version 431 • 6 months ago • Options • Reply

^ 0 ▼

Thanks



Nabil Jabareen • Posted on Version 438 • 6 months ago • Options • Reply

^ 0 ▼

Thanks for sharing! A lot to learn right here!!

DATAI



DATAI Kernel Author • Posted on Version 442 • 6 months ago • Options • Reply

^ 0 ▼

You are welcome



Nirjar • Posted on Version 442 • 6 months ago • Options • Reply

^ 0 ▼

Thanks for this great Notebook, really helpful !!

DATAI



DATAI Kernel Author • Posted on Version 442 • 6 months ago • Options • Reply

^ 0 ▼

Thank you for your comment.



Kunal Katdare • Posted on Version 442 • 6 months ago • Options • Reply

^ 0 ▼

Thanks for the notebook!

DATAI

DATAI Kernel Author

• Posted on Version 452 • 6 months ago • Options • Reply

^ 0 ▼



You are welcome.



Sofie Michiels • Posted on Version 443 • 6 months ago • Options • Reply

^ 0 ▼

Nice summary of useful data manipulation functionality with Python - thank you!

DATAI

DATAI Kernel Author

• Posted on Version 474 • 6 months ago • Options • Reply

^ 0 ▼



You are welcome



TanthiamhuatTan • Posted on Version 443 • 6 months ago • Options • Reply

^ 0 ▼

```
print(check_output(["ls", "../input"]).decode("utf8"))
is the above command only applicable for the Linux system? how to modify for the Windows system?
```



Drain • Posted on Version 462 • 6 months ago • Options • Reply

^ 0 ▼

great notebook

DATAI

DATAI Kernel Author

• Posted on Version 483 • 6 months ago • Options • Reply

^ 0 ▼



You are welcome



Yi Xiang • Posted on Version 465 • 6 months ago • Options • Reply

^ 0 ▼

Really helpful notebook!

DATAI

DATAI Kernel Author

• Posted on Version 474 • 6 months ago • Options • Reply

^ 0 ▼



You are welcome

SamuelSperling • Posted on Version 472 • 6 months ago • Options • Reply

^ 0 ▼



Thanks for putting this together!! :)

DATAI

DATAI Kernel Author

• Posted on Version 474 • 6 months ago • Options • Reply

^ 0 v

You are welcome



r1chard • Posted on Version 473 • 6 months ago • Options • Reply

^ 0 v

Thanks for sharing!

DATAI

DATAI Kernel Author

• Posted on Version 474 • 6 months ago • Options • Reply

^ 0 v

You are welcome



Djane • Posted on Version 474 • 6 months ago • Options • Reply

^ 0 v

Really helpful for folks like me who has no clue

DATAI

DATAI Kernel Author

• Posted on Version 474 • 6 months ago • Options • Reply

^ 0 v

Thank you for your comment.



Dobby • Posted on Version 475 • 6 months ago • Options • Reply

^ 0 v

Thanks

DATAI

DATAI Kernel Author

• Posted on Version 483 • 6 months ago • Options • Reply

^ 0 v

You are welcome



gstaxy • Posted on Version 475 • 6 months ago • Options • Reply

^ 0 v

Very helpful tutorial! Thanks.

DATAI

DATAI Kernel Author

• Posted on Version 483 • 6 months ago • Options • Reply

^ 0 v

You are welcome



David • Posted on Version 479 • 6 months ago • Options • Reply

^ 0 ▼

I wish I had this when I first started learning, great work!

DATAI

DATAI Kernel Author

• Posted on Version 483 • 6 months ago • Options • Reply

^ 0 ▼



Thank you for your comment.



hodgesz • Posted on Version 480 • 6 months ago • Options • Reply

^ 0 ▼

thanks

DATAI

DATAI Kernel Author

• Posted on Version 483 • 6 months ago • Options • Reply

^ 0 ▼



You are welcome



-Maran • Posted on Version 483 • 6 months ago • Options • Reply

^ 0 ▼

Thanks a lot. Very helpful for beginners . I really enjoyed it.

DATAI

DATAI Kernel Author

• Posted on Version 483 • 6 months ago • Options • Reply

^ 0 ▼



Thanks



-Maran • Posted on Version 483 • 6 months ago • Options • Reply

^ 0 ▼



BondidSg • Posted on Version 491 • 6 months ago • Options • Reply

^ 0 ▼

Hey guys im a beginner in ML, im actually confused whether i should go for R or Python please help me.

DATAI

DATAI Kernel Author

• Posted on Version 491 • 6 months ago • Options • Reply

^ 0 ▼



Python



Kerem Sahin • Posted on Version 492 • 6 months ago • Options • Reply

^ 0 ▾

Good to see this as one of the top 5 kernels. (Tebrikler!)

DATAI

DATAI Kernel Author

• Posted on Version 493 • 6 months ago • Options • Reply

^ 0 ▾



Thank you for your comment.(tesekkurer)



Will Campbell • Posted on Version 493 • 6 months ago • Options • Reply

^ 0 ▾

Wow thank you for this! great resource!

DATAI

DATAI Kernel Author

• Posted on Version 493 • 6 months ago • Options • Reply

^ 0 ▾



You are welcome



KrishnaK • Posted on Version 493 • 6 months ago • Options • Reply

^ 0 ▾

Thanks for such informative article. I am new to python. And your article helped a lot to get started. Will surely keep coming back for reference.

DATAI

DATAI Kernel Author

• Posted on Version 493 • 6 months ago • Options • Reply

^ 0 ▾



Thank you for your comment.



LocLuong • Posted on Version 514 • 6 months ago • Options • Reply

^ 0 ▾

Nice kernel

DATAI

DATAI Kernel Author

• Posted on Version 514 • 6 months ago • Options • Reply

^ 0 ▾



Thanks



LocLuong • Posted on Version 514 • 6 months ago • Options • Reply

^ 0 ▾

Thanks you

Srihari Katti • Posted on Version 524 • 6 months ago • Options • Reply

^ 0 ▾



For a dilettante in Data science this Notebook is a bible

DATAI

DATAI Kernel Author

• Posted on Version 526 • 6 months ago • Options • Reply

^ 0 ▾

Thank you for your comment.



Md khirul islam • Posted on Version 525 • 6 months ago • Options • Reply

^ 0 ▾

This is a great intro for beginners . Thank you so much

DATAI

DATAI Kernel Author

• Posted on Version 526 • 6 months ago • Options • Reply

^ 0 ▾

you are welcome



Amine Hy • Posted on Version 531 • 6 months ago • Options • Reply

^ 0 ▾

This is a good place to start for beginners

DATAI

DATAI Kernel Author

• Posted on Version 533 • 6 months ago • Options • Reply

^ 0 ▾

Thank you for your comment.



gcoolt • Posted on Version 531 • 6 months ago • Options • Reply

^ 0 ▾

Thanks for taking the time to write this up. Very helpful

DATAI

DATAI Kernel Author

• Posted on Version 533 • 6 months ago • Options • Reply

^ 0 ▾

You are welcome



Arun Godwin Patel • Posted on Version 536 • 6 months ago • Options • Reply

^ 0 ▾

Really nice work, I will use some of these techniques for sure!

DATAI

DATAI Kernel Author

• Posted on Version 536 • 6 months ago • Options • Reply

^ 0 ▾

Thank you for your comment



Atri Saxena • Posted on Version 537 • 6 months ago • Options • Reply

^ 0 ▼

Thank you for this wonderful notebook.

DATAI

DATAI Kernel Author

• Posted on Version 538 • 6 months ago • Options • Reply

^ 0 ▼



Thank you for your all comments :)



Sarzhann • Posted on Version 555 • 5 months ago • Options • Reply

^ 0 ▼

Good Job!

DATAI

DATAI Kernel Author

• Posted on Version 555 • 5 months ago • Options • Reply

^ 0 ▼



Thank you



Arvind Kumar • Posted on Version 562 • 5 months ago • Options • Reply

^ 0 ▼

can any one tell me element of statisticsal learning book is worth reading or not?????????

DATAI

DATAI Kernel Author

• Posted on Version 562 • 5 months ago • Options • Reply

^ 0 ▼



Check elements of statistical learning online course from stanford university you will find what you need book or other materials



Arvind Kumar • Posted on Version 562 • 5 months ago • Options • Reply

^ 0 ▼

great work...

DATAI

DATAI Kernel Author

• Posted on Version 565 • 5 months ago • Options • Reply

^ 0 ▼



Thanks



Sujit Gupta • Posted on Version 562 • 5 months ago • Options • Reply

^ 0 ▼

Thanks for sharing !!

DATAI

DATAI Kernel Author

• Posted on Version 565 • 5 months ago • Options • Reply

^ 0 ▼

You are welcome



Siva • Posted on Version 562 • 5 months ago • Options • Reply

^ 0 ▼

Just joined Kaggle recently and found this tutorial very useful..thank you :)

DATAI

DATAI Kernel Author

• Posted on Version 565 • 5 months ago • Options • Reply

^ 0 ▼

You are welcome



Sudeep Shukla • Posted on Version 566 • 5 months ago • Options • Reply

^ 0 ▼

This notebook is a gold mine, using it as a reference all the time. Thanks a lot!

DATAI

DATAI Kernel Author

• Posted on Version 566 • 5 months ago • Options • Reply

^ 0 ▼

You are welcome



Harish • Posted on Version 566 • 5 months ago • Options • Reply

^ 0 ▼

It is really awesome

DATAI

DATAI Kernel Author

• Posted on Version 566 • 5 months ago • Options • Reply

^ 0 ▼

Thank you for your comment



Sushil Nath Gupta • Posted on Version 569 • 5 months ago • Options • Reply

^ 0 ▼

Absolute notebook for beginners !! Clear and concise !!

DATAI

DATAI Kernel Author

• Posted on Version 570 • 5 months ago • Options • Reply

^ 0 ▼

Thank you for your comment.



Minh-Thanh Kho... • Posted on Version 587 • 5 months ago • Options • Reply

^ 0 ▼

Great introduction! Thank you!

DATAI

DATAI Kernel Author

• Posted on Version 588 • 5 months ago • Options • Reply

^ 0 ▼

•••• You are welcome



William Carlos • Posted on Version 597 • 5 months ago • Options • Reply

^ 0 ▼

Thanks Kaan!!! Congratulations, it helped a lot.

DATAI

DATAI Kernel Author

• Posted on Version 611 • 5 months ago • Options • Reply

^ 0 ▼

•••• Thanks William for your comment.



Chen_shengabc • Posted on Version 602 • 5 months ago • Options • Reply

^ 0 ▼

does have any other program language

DATAI

DATAI Kernel Author

• Posted on Version 611 • 5 months ago • Options • Reply

^ 0 ▼

•••• Like R?



Bassel Matroud • Posted on Version 604 • 5 months ago • Options • Reply

^ 0 ▼

thank you

DATAI

DATAI Kernel Author

• Posted on Version 611 • 5 months ago • Options • Reply

^ 0 ▼

•••• You are welcome



Prithviraj Jadhav • Posted on Version 604 • 5 months ago • Options • Reply

^ 0 ▼

really awesome notebook for noobs like me

DATAI

DATAI Kernel Author • Posted on Version 611 • 5 months ago • Options • Reply

 0 

Thank you for your comment.



JacquesBikoundo... • Posted on Version 605 • 5 months ago • Options • Reply

 0 

Very useful. It looks like my python Bible!

DATAI

DATAI Kernel Author • Posted on Version 611 • 5 months ago • Options • Reply

 0 

Thanks.



Jack Lee • Posted on Version 614 • 5 months ago • Options • Reply

 0 

Great work! Thank you!!:)

DATAI

DATAI Kernel Author • Posted on Version 619 • 5 months ago • Options • Reply

 0 

You are welcome



Soumya Saha • Posted on Version 619 • 5 months ago • Options • Reply

 0 

Nice. Detailed notebook. Thank you.

DATAI

DATAI Kernel Author • Posted on Version 619 • 5 months ago • Options • Reply

 0 

You are welcome



Gou Feng • Posted on Version 633 • 5 months ago • Options • Reply

 0 

great guide for beginners Thanks

DATAI

DATAI Kernel Author • Posted on Version 635 • 5 months ago • Options • Reply

 0 

You are welcome



Haytam Zanid • Posted on Version 641 • 4 months ago • Options • Reply

^ 0 v

Thank you for the awesome tutorial!

DATAI

DATAI Kernel Author



• Posted on Version 656 • 4 months ago • Options • Reply

^ 0 v

Check my other tutorials like deep learning, i think you will like them too



Sidharth Menon • Posted on Version 650 • 4 months ago • Options • Reply

^ 0 v

Thanks a lot Kaan! Looking forward to learning a lot with this. :D

DATAI

DATAI Kernel Author



• Posted on Version 656 • 4 months ago • Options • Reply

^ 0 v

You are welcome :)



Shyamprasad NA... • Posted on Version 650 • 4 months ago • Options • Reply

^ 0 v

Thanks Kaan, for your time in preparing valuable notebook for beginners like me. Keep it up the good work.

DATAI

DATAI Kernel Author



• Posted on Version 656 • 4 months ago • Options • Reply

^ 0 v

You check my other tutorials right? I will make :) thank you for your comment



Yokesh • Posted on Version 651 • 4 months ago • Options • Reply

^ 0 v

Thanks, Can, I would start participating in to competitions more confidently.

DATAI

DATAI Kernel Author



• Posted on Version 656 • 4 months ago • Options • Reply

^ 0 v

I glad to hear that. Just try... thank you for your comment



welkin • Posted on Version 654 • 4 months ago • Options • Reply

^ 0 v

This organized resource is much appreciated. Thanks.

DATAI

DATAI Kernel Author

• Posted on Version 656 • 4 months ago • Options • Reply

^ 0 ▼

You are welcome



Glassof Water • Posted on Version 656 • 4 months ago • Options • Reply

^ 0 ▼

Hi Kaan, great tutorial, very helpful. I had one question. When dropping NaN values you use: data1 = data
data1["Type 2"].dropna(inplace = True) This seems to also drop the NaN values in data. Why is this the case? How
would one change data1 without changing data?

DATAI

DATAI Kernel Author

• Posted on Version 656 • 4 months ago • Options • Reply

^ 0 ▼

I want to drop all nan values to explain concepts better. If you make inplace = false, you do not
change data. Thank you for your question and comment



Glassof Water • Posted on Version 659 • 4 months ago • Options • Reply

^ 0 ▼

inplace = false

Great! Thanks for your response



sirivella madhu • Posted on Version 660 • 4 months ago • Options • Reply

^ 0 ▼

Thank you Kann.. create tutorial for beginners. :)

DATAI

DATAI Kernel Author

• Posted on Version 660 • 4 months ago • Options • Reply

^ 0 ▼

You are welcome



Zootojia • Posted on Version 662 • 4 months ago • Options • Reply

^ 0 ▼

so far the best data science tutorial I have ever seen on Kaggle. Thanks very much for sharing .

DATAI

DATAI Kernel Author

• Posted on Version 669 • 4 months ago • Options • Reply

^ 0 ▼

Thank you for your comment



Bruno Batista • Posted on Version 668 • 4 months ago • Options • Reply

^ 0 ▼

Great ! Im starting data science! That helps a lot !Thanks man!

DATAI

DATAI Kernel Author

• Posted on Version 669 • 4 months ago • Options • Reply

^ 0 ▼

You are welcome

• • Posted on Version 669 • 4 months ago • Options • Reply

^ 0 ▼

• • Posted on Version 669 • 4 months ago • Options • Reply

^ 0 ▼

• • Posted on Version 669 • 4 months ago • Options • Reply

^ 0 ▼

DATAI

DATAI Kernel Author

• Posted on Version 669 • 4 months ago • Options • Reply

^ 0 ▼

This time i write comment before you deleting :) thank you for your comment



I love the norther... • Posted on Version 670 • 4 months ago • Options • Reply

^ 0 ▼

Thank you,I am a beginner

DATAI

DATAI Kernel Author

• Posted on Version 670 • 4 months ago • Options • Reply

^ 0 ▼

I glad you like it



I love the norther... • Posted on Version 670 • 4 months ago • Options • Reply

^ 0 ▼



I love the norther... • Posted on Version 670 • 4 months ago • Options • Reply

^ 0 ▼



I love the norther... • Posted on Version 670 • 4 months ago • Options • Reply

^ 0 ▼



I love the norther... • Posted on Version 670 • 4 months ago • Options • Reply

^ 0 ▼



Mateo • Posted on Version 673 • 4 months ago • Options • Reply

^ 0 ▼

Great tutorial for beginners

DATAI

DATAI Kernel Author

•••••

Thank you for your comment

^ 0 ▼



BlastChar • Posted on Version 674 • 4 months ago • Options • Reply

^ 0 ▼

Very nice and very useful. Great work

DATAI

DATAI Kernel Author

•••••

Thanks.

^ 0 ▼



Mudit Tiwari • Posted on Version 681 • 4 months ago • Options • Reply

^ 0 ▼

Thankyou Kaan!! This notebook was really helpful for me.

DATAI

DATAI Kernel Author

•••••

You are welcome

^ 0 ▼



Mudit Tiwari • Posted on Version 681 • 4 months ago • Options • Reply

^ 0 ▼



Hemasundar Rao • Posted on Version 692 • 4 months ago • Options • Reply

^ 0 ▼

thanks .. i have learned so many things it is very useful to beginners

DATAI

DATAI Kernel Author

•••••

You are welcome

^ 0 ▼



Vish B • Posted on Version 692 • 4 months ago • Options • Reply

^ 0 ▼

Perfect tutorial for a beginner like myself. On to the next one. :D

DATAI

DATAI Kernel Author

• Posted on Version 694 • 4 months ago • Options • Reply

^ 0 ▼

•••••
Thank you for your comment



Bartho PAMEN • Posted on Version 692 • 4 months ago • Options • Reply

^ 0 ▼

Precise and straight forward for beginners. Thanks !!

DATAI

DATAI Kernel Author

• Posted on Version 694 • 4 months ago • Options • Reply

^ 0 ▼

•••••
I glad you like it.



Tommaso Mariani • Posted on Version 696 • 4 months ago • Options • Reply

^ 0 ▼

Beautiful tutorial!

DATAI

DATAI Kernel Author

• Posted on Version 697 • 4 months ago • Options • Reply

^ 0 ▼

•••••
Thanks



giasuddin • Posted on Version 697 • 4 months ago • Options • Reply

^ 0 ▼

Thank you,very helpful.

DATAI

DATAI Kernel Author

• Posted on Version 697 • 4 months ago • Options • Reply

^ 0 ▼

•••••
You are welcome



Vivek kumar • Posted on Version 701 • 4 months ago • Options • Reply

^ 0 ▼

Worth Reading

DATAI

DATAI Kernel Author

• Posted on Version 702 • 4 months ago • Options • Reply

 0 

Thanks



Nico • Posted on Version 702 • 4 months ago • Options • Reply

 0 

Thanks for sharing. It will be my main reference. :)

DATAI

DATAI Kernel Author

• Posted on Version 704 • 4 months ago • Options • Reply

 0 

You are welcome



Md Raihan Sharif • Posted on Version 704 • 4 months ago • Options • Reply

 0 

It's really useful for the beginners. Thanks a lot for sharing your ideas.

DATAI

DATAI Kernel Author

• Posted on Version 704 • 4 months ago • Options • Reply

 0 

Thanks for your comment



Sahil Mani • Posted on Version 705 • 4 months ago • Options • Reply

 0 

I learned a lot. Thanks for makein gthis kernel

DATAI

DATAI Kernel Author

• Posted on Version 705 • 4 months ago • Options • Reply

 0 

You are welcome



Mulder • Posted on Version 706 • 4 months ago • Options • Reply

 0 

The tutorial is quite intuitive and easy learning for beginners, truly appreciate your efforts.

DATAI

DATAI Kernel Author

• Posted on Version 706 • 4 months ago • Options • Reply

 0 

Thank you



Martin H • Posted on Version 707 • 4 months ago • Options • Reply

^ 0 ▼

Really appreciate your excellent tutorial!

DATAI

DATAI Kernel Author

• Posted on Version 707 • 4 months ago • Options • Reply

^ 0 ▼



Thank you for your comment



Quan Nguyen • Posted on Version 707 • 4 months ago • Options • Reply

^ 0 ▼

Thank you, this must be my Data Science starting note book!

DATAI

DATAI Kernel Author

• Posted on Version 707 • 4 months ago • Options • Reply

^ 0 ▼



Yes it is :) thank you for your comment



Tim Reeder • Posted on Version 714 • 4 months ago • Options • Reply

^ 0 ▼

What an entertaining and educative notebook for beginners, thanks for this :)

DATAI

DATAI Kernel Author

• Posted on Version 717 • 4 months ago • Options • Reply

^ 0 ▼



You are welcome



Junying (Emma) ... • Posted on Version 715 • 4 months ago • Options • Reply

^ 0 ▼

thanks, very helpful

DATAI

DATAI Kernel Author

• Posted on Version 717 • 4 months ago • Options • Reply

^ 0 ▼



Thanks for your comment



Khairul Islam • Posted on Version 721 • 4 months ago • Options • Reply

^ 0 ▼

wow , very useful one

DATAI

DATAI Kernel Author

• Posted on Version 723 • 3 months ago • Options • Reply

^ 0 ▼



Thanks



Harun-Ur-Rashid(... • Posted on Version 725 • 3 months ago • Options • Reply

^ 0 ▼

Awesome kernel...But i need some help from you....

DATAI



DATAI Kernel Author • Posted on Version 727 • 3 months ago • Options • Reply

^ 0 ▼

How can i help you. Thanks for your comment



Jeff Golden • Posted on Version 726 • 3 months ago • Options • Reply

^ 0 ▼

Thank you for sharing -- I'm new to Kaggle and new to Python so this will be a big help!

DATAI



DATAI Kernel Author • Posted on Version 727 • 3 months ago • Options • Reply

^ 0 ▼

You are welcome check my other kernels too.



greg • Posted on Version 726 • 3 months ago • Options • Reply

^ 0 ▼

Thanks a lot ! As a beginner, I find this Kernel very useful and the examples are neat. Congrats

DATAI



DATAI Kernel Author • Posted on Version 727 • 3 months ago • Options • Reply

^ 0 ▼

Thank you so much for your comment.



Muammer Hüseyi... • Posted on Version 727 • 3 months ago • Options • Reply

^ 0 ▼

Masterpiece! (Eline sağlık hocam)

DATAI



DATAI Kernel Author • Posted on Version 727 • 3 months ago • Options • Reply

^ 0 ▼

Thanks(faydalı olduysa ne mutlu)



Adeel Akram • Posted on Version 729 • 3 months ago • Options • Reply

^ 0 ▼

Thanks for sharing, useful kernel indeed.

DATAI

DATAI Kernel Author

• Posted on Version 732 • 3 months ago • Options • Reply

^ 0 ∨

Thank you for your comment



George Haglund • Posted on Version 730 • 3 months ago • Options • Reply

^ 0 ∨

Great

DATAI

DATAI Kernel Author

• Posted on Version 732 • 3 months ago • Options • Reply

^ 0 ∨

I glad you like it



Sheikh Nazmul Is... • Posted on Version 730 • 3 months ago • Options • Reply

^ 0 ∨

Thank you so much!

DATAI

DATAI Kernel Author

• Posted on Version 732 • 3 months ago • Options • Reply

^ 0 ∨

You are welcome



Akash Kandpal • Posted on Version 731 • 3 months ago • Options • Reply

^ 0 ∨

many many thanks Sir :)

DATAI

DATAI Kernel Author

• Posted on Version 732 • 3 months ago • Options • Reply

^ 0 ∨

You are welcome Akash



Rachit Saxena • Posted on Version 732 • 3 months ago • Options • Reply

^ 0 ∨

Thanks a lot @kanncaa1. As a beginner, I found it really helpful.

DATAI

DATAI Kernel Author

• Posted on Version 732 • 3 months ago • Options • Reply

^ 0 ∨

Thank you for your comment.



fjones • Posted on Version 733 • 3 months ago • Options • Reply

^ 0 ▼

This is very helpful, thanks!!

DATAI

DATAI Kernel Author

• Posted on Version 734 • 3 months ago • Options • Reply

^ 0 ▼

• • • •

Thank you for your comment

• Posted on Version 733 • 3 months ago • Options • Reply

^ 0 ▼



Maurice • Posted on Version 734 • 3 months ago • Options • Reply

^ 0 ▼

wow, it is very very interesting and useful to beginners in Data Science. Thanks ;)

DATAI

DATAI Kernel Author

• Posted on Version 738 • 3 months ago • Options • Reply

^ 0 ▼

• • • •

You are welcome

Tapas • Posted on Version 735 • 3 months ago • Options • Reply

^ 0 ▼

Very nice tutorial, to know basic about the data science

DATAI

DATAI Kernel Author

• Posted on Version 738 • 3 months ago • Options • Reply

^ 0 ▼

• • • •

Thanks

KraLMachine • Posted on Version 738 • 3 months ago • Options • Reply

^ 0 ▼

Thanks for your sharing.



Naveen • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Excellent kernel. Great work to share the thought for beginners.

DATAI

DATAI Kernel Author

• Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

• • • •

Thanks



hasannajeeb • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Thnaks alot Kan. Its very well written and easy tutorial



KZKSY • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Good note book. Just in time for my upcoming project.



KZKSY • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼



omer dogru • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Great work dude.



ahmetserguns • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Excellent work.



Andrew • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

To put in polite terms, this (and your other pages) absolutely poos on the Uni workshops I'm getting at the moment. I have been directing others here, and cannot speak highly enough of the offerings you have presented.



Nimoh • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

It really helped me to understand data science; i am absolute beginner in python and data science and it really helped me. thank you



eray kisabacak • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Excellent work. Thanks Kaan!!



Abdullah BARHA • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Your udemy courses about python, and your kernels are very helpful for beginners. Thank you.



Kubilay Çiftci • Posted on Version 739 • 2 months ago • Options • Reply

^ 0 ▼

Great Notebook for beginners, like me. Thanks for your work :)

• Posted on Latest Version • 2 months ago • Options • Reply

^ 0 ▼



Abdullah ALAN • Posted on Latest Version • 2 months ago • Options • Reply

^ 0 ▼

thanks for your support

• Posted on Latest Version • 2 months ago • Options • Reply

^ 0 ▼



Dede • Posted on Latest Version • 2 months ago • Options • Reply

^ 0 ▼

This is clear and detailed for the beginner level. Thank you.



Fırat Doğan • Posted on Latest Version • 2 months ago • Options • Reply

^ 0 ▼

It is helpful and brief.



spektral • Posted on Latest Version • 2 months ago • Options • Reply

^ 0 ▼

nice work



Selamet Şamli • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

Great job! Very comprehensive and helpful, thanks.



Tevfik Bagcivan • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

This tutorial also is over. Thanks Kaan. I will see you in the next tutorial.



SerdarA • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

I have just started to work with this kernel, it is very well structured and a good step by step guideline for beginners.



mafabi emmanuel • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

this kernel was so fruitful thank you

Volkan Sahin • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼



Thank you for this marvelous work !



Mehmet Icme • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

Kaan hocam elinize sağlık. Çok güzel çalışma olmuş.

•

• Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼



kmluns • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

That job is so good. I'm watching your videos on Udemy. Thank you, and keep it up!



Tuna Han • Posted on Latest Version • a month ago • Options • Reply

^ 0 ▼

This was really helpful for a beginner like me. Thanks a lot. Keep up the great work!!!



Ievent • Posted on Latest Version • 25 days ago • Options • Reply

^ 0 ▼

Thank's Kaan, data review and educational information is very nice.



Ebubekir Aygün • Posted on Latest Version • 24 days ago • Options • Reply

^ 0 ▼

Thanks Kaan, i learned alot from you.



Berkay GUNGOR • Posted on Latest Version • 24 days ago • Options • Reply

^ 0 ▼

Thank you for this instructive kernel. Without it i would not be able to start learning phyton and data science so fast.



Wilson Lúcio • Posted on Latest Version • 23 days ago • Options • Reply

^ 0 ▼

Great Notebook!



Harun-Ur-Rashid(...) • Posted on Latest Version • 23 days ago • Options • Reply

^ 0 ▼

WoW ! Great work ! kindly check my new [kernel](#) and give feedback or if you like this kernel give upvote.Advance thanks.



Kerem ESER • Posted on Latest Version • 22 days ago • Options • Reply

^ 0 ▼

Thanks for you awesome work.



Safak • Posted on Latest Version • 20 days ago • Options • Reply

^ 0 ▼

Thank you for this lesson



MuratUSLU • Posted on Latest Version • 19 days ago • Options • Reply

^ 0 ▼

`data[np.logical_and(data['Defense']>200, data['Attack']>100)]` I have tried to add additional filter to code above but i am getting an error message , would you comment whats wrong. Thanks

`data[np.logical_and(data['Defense']>200, data['Attack']>100, data ["Speed"]>40)]`

TypeError Traceback (most recent call last) in () 1 # 2 - Filtering pandas with logical_and 2 # There are only 2 pokemons who have higher defence value than 200 and higher attack value than 100 ----> 3 `data[np.logical_and(data['Defense']>200, data['Attack']>100, data ["Speed"]>40)]`

TypeError: return arrays must be of ArrayType



Nasir Islam Suj... • Posted on Latest Version • 17 days ago • Options • Reply

^ 0 ▼

`numpy.logical_and` only takes two input parameters, the third if passed is assumed to be an output array. That's why you are getting the TypeError.

Use single & instead:

`data[(data['Defense']>200) & (data['Attack']>100) & (data['Speed']>40)]`



Abhishek • Posted on Latest Version • 19 days ago • Options • Reply

^ 0 ▼

Amazing job, thankyou!



Osman Işık • Posted on Latest Version • 18 days ago • Options • Reply

^ 0 ▼

This project very helpful and useful. Thank you for this. I wanna see your new kernels :)



Nasir Islam Sujan • Posted on Latest Version • 17 days ago • Options • Reply

^ 0 ▼

One of the great tutorial i have seen so far to start with. Thanks a lot.



Gokhan Ince • Posted on Latest Version • 16 days ago • Options • Reply

^ 0 ▼

Great course :)



Thanks Kaan, this notebook was very helpful to me.



Anton V S • Posted on Latest Version • 13 days ago • Options • Reply

^ 0 ▼

Hi! Could you please give some approximate estimations on how much time does it take to complete this tutorial? Please consider a man having basic skills in programming. Thanks!



daresnick • Posted on Latest Version • 13 days ago • Options • Reply

^ 0 ▼

Thanks. Good summary of useful python code for data scientists.



Samet BOR • Posted on Latest Version • 13 days ago • Options • Reply

^ 0 ▼

Thank you for your tutorial.



Emre Can Çelik • Posted on Latest Version • 10 days ago • Options • Reply

^ 0 ▼

I think this kernel is very useful everyone who wants to learning data science.



cvarlist13 • Posted on Latest Version • 10 days ago • Options • Reply

^ 0 ▼

Wonderful tutorial! Thank you so much



Asmin • Posted on Latest Version • 8 days ago • Options • Reply

^ 0 ▼

Thanks for your effort. It is helpfull and easy to follow.



Ahmet CAN • Posted on Latest Version • 7 days ago • Options • Reply

^ 0 ▼

thanks teacher, your narration and examples are very good



Alihan Tabak • Posted on Latest Version • 6 days ago • Options • Reply

^ 0 ▼

Amazingly great!



Mateusz Czerwiń... • Posted on Latest Version • 6 days ago • Options • Reply

^ 0 ▼

I would like to say thank you. This kernel helps me a lot and show the example path how to start solving problems.



Dickie Greenleaf • Posted on Latest Version • 6 days ago • Options • Reply

^ 0 ▼

Exceptional



Mustafa CAN • Posted on Latest Version • a day ago • Options • Reply

^ 0 ▼

Thanks for everything. I am moving on.

Similar Kernels



Titanic Survival
Prediction End To End
ML Pipeline



Predicting Flight Delays
[Tutorial]



Home Credit :
Complete EDA +
Feature Importance ✓✓



Home Credit Default
Risk Extensive EDA



Plotly Tutorial For
Beginners

© 2018 Kaggle Inc

[Our Team](#) [Terms](#) [Privacy](#) [Contact/Support](#)

