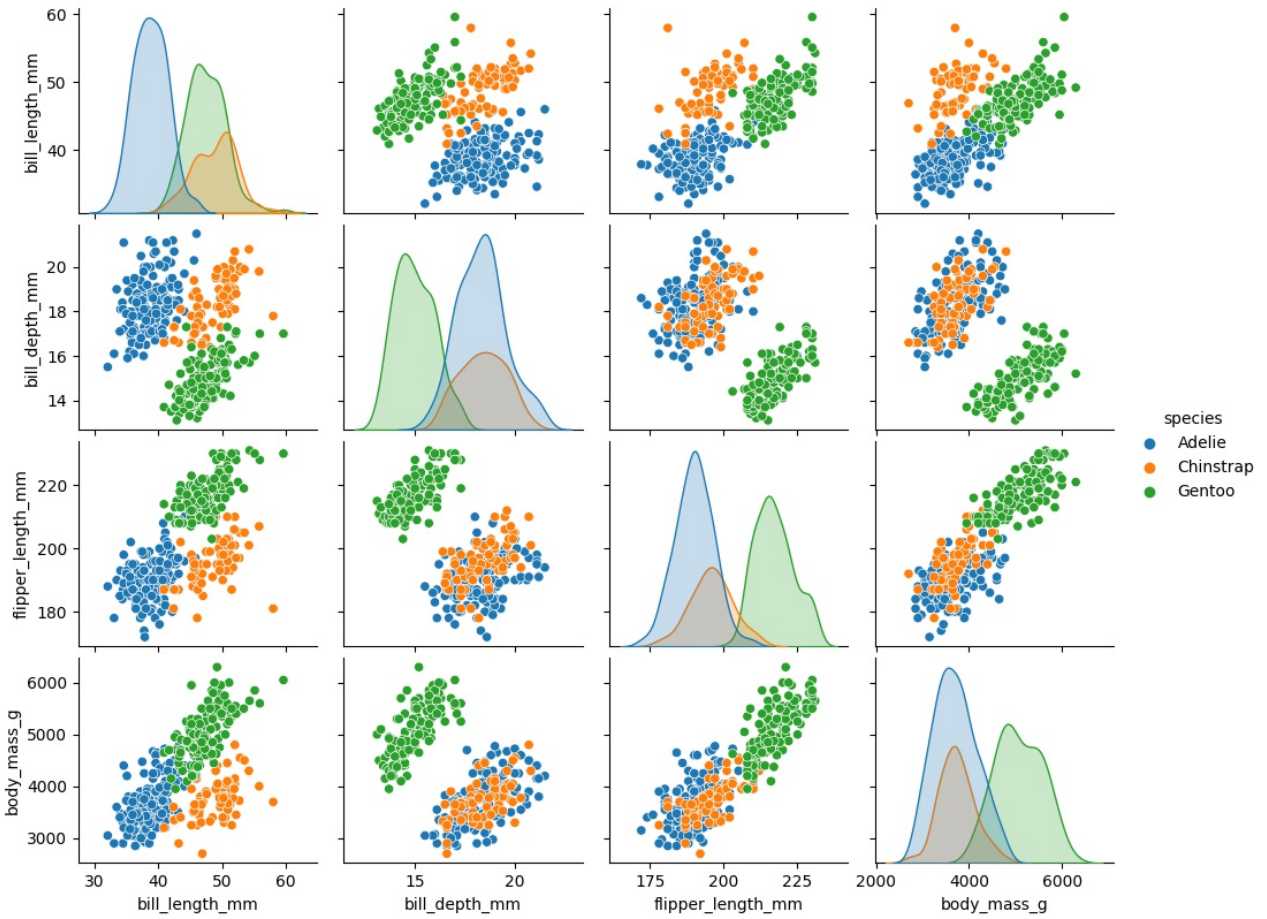
Ver. BPM
Status!
Author MK

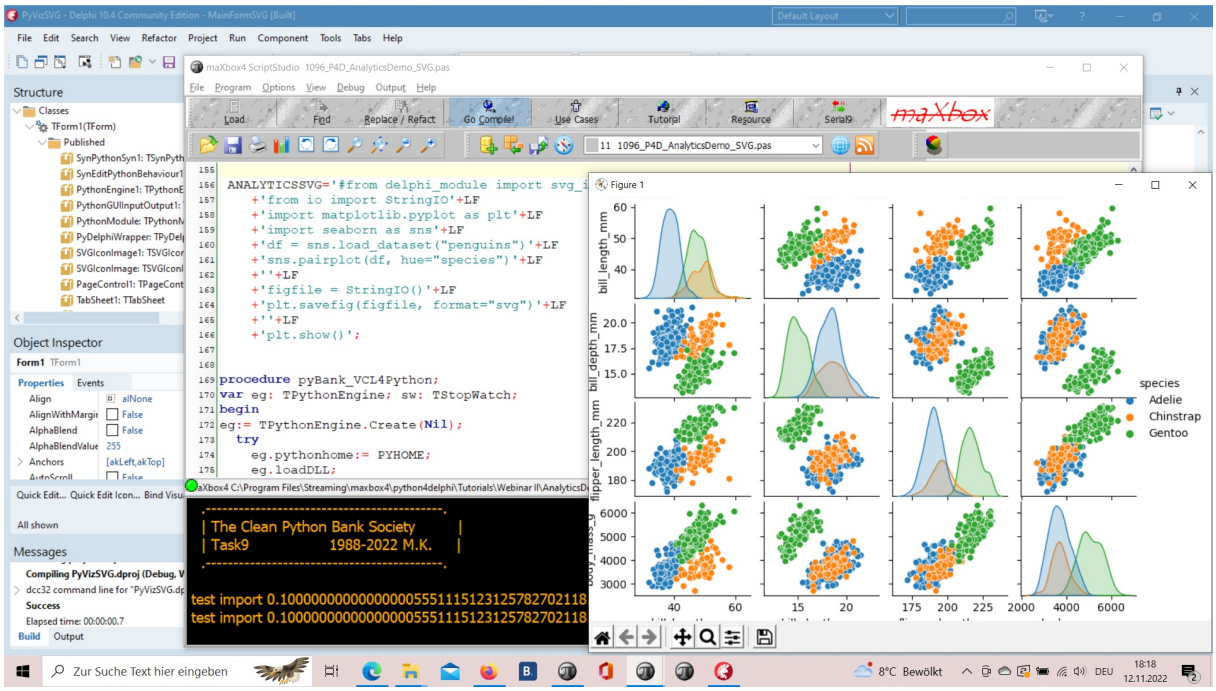
Penguin Data Science

maXbox Starter 101 P4D



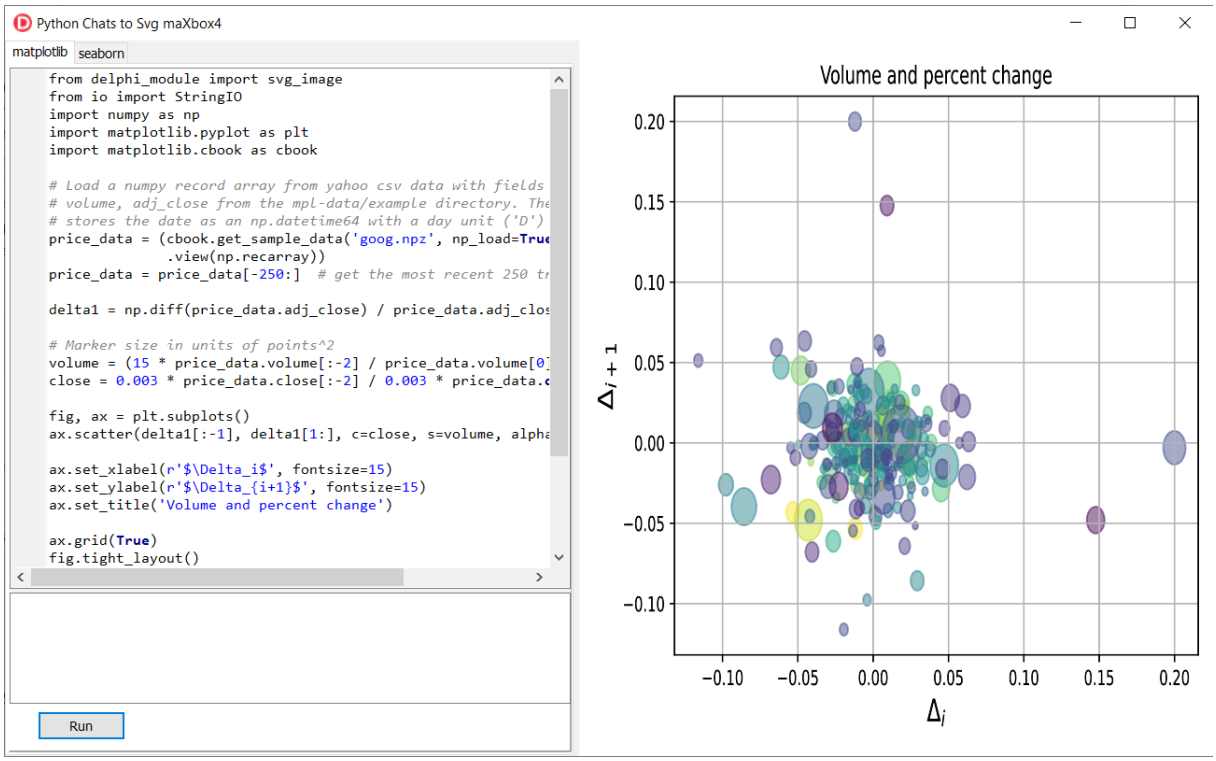
Links and Sources
Title
https://www.kaggle.com/code/parulpandey/penguin-dataset-the-new-iris
python4delphi/Tutorials/Webinar II at master · maxkleiner/python4delphi (github.com)

https://maxbox4.wordpress.com/



1096_2022-11-12_svg_delphi_maxbox_seaborn4.png

RAD Studio 11.4, maXbox4 and Python 3.8 with Seaborn



Python4Delphi with SVG Plot

1 From Data to Plot

1.1 Data Exploration

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper.

In this paper I want to show a machine learning lecture with Python4Delphi. The Palmer Archipelago (Antarctica) penguin dataset appears to be a drop in replacement for the same. It is a great intro dataset for data exploration & visualization. But the penguins dataset has different number of samples for each species. It can be observed that unlike the Iris dataset, this data contains different number of entries for each specie.

The template for the most P4D programs goes like this:

```
with TpythonEngine.Create(Nil) do begin
  pythonhome:= PYHOME;
  try
    loadDLL;
    Println('Decimal: '+
          EvalStr('__import__("decimal").Decimal(0.1)'));
  except
    raiseError;
  finally
    free;
  end;
end;
```

The minimal configuration depends on your Python-installation and the *UseLastKnownVersion* property in *TDynamicDll* but once known it goes also with *raiseError* to catch the Python exceptions In your code. For the const of PYHOME on my machine it depends on the version:

Const PYHOME = 'C:\Users\max\AppData\Local\Programs\Python\Python36-32\';
PYHOME = 'C:\Users\max4\AppData\Local\Programs\Python\Python37-32\';

From code to a short introduction to the shape of a penguin s head with his culmen.
The upper margin of the beak or bill is referred to as the culmen and the measurement is taken using calipers with one jaw at the tip of the upper mandible and the other at base of the skull or the first feathers depending on the standard chosen.

First we get the data with seaborn:

```
import seaborn as sns
df = sns.load_dataset("penguins")
```

Or alternate sources

```
df = pd.read_csv('../input/palmer-archipelago-antarctica-penguin-data/penguins_size.csv')
```

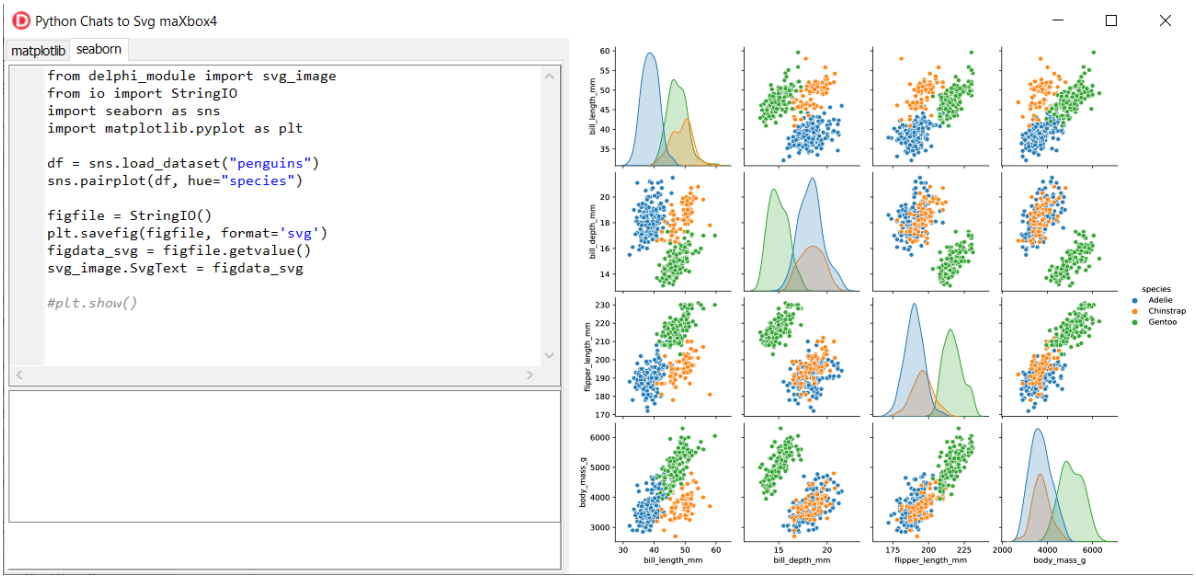
```
p2data =
"https://raw.githubusercontent.com/allisonhorst/palmerpenguins/master/data-raw/penguins_raw.csv
p2data =  https://raw.githubusercontent.com/rianrajagede/penguin-python/master/Datasets/penguins.csv
```

```
df.head()
```

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
count	344	344	342.000000	342.000000	342.000000	342.000000	334
unique	3	3	NaN	NaN	NaN	NaN	3
top	Adelie	Biscoe	NaN	NaN	NaN	NaN	MALE
freq	152	168	NaN	NaN	NaN	NaN	168
mean	NaN	NaN	43.921930	17.151170	200.915205	4201.754386	NaN

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
std	NaN	NaN	5.459584	1.974793	14.061714	801.954536	NaN
min	NaN	NaN	32.100000	13.100000	172.000000	2700.000000	NaN
25%	NaN	NaN	39.225000	15.600000	190.000000	3550.000000	NaN
50%	NaN	NaN	44.450000	17.300000	197.000000	4050.000000	NaN
75%	NaN	NaN	48.500000	18.700000	213.000000	4750.000000	NaN
max	NaN	NaN	59.600000	21.500000	231.000000	6300.000000	NaN



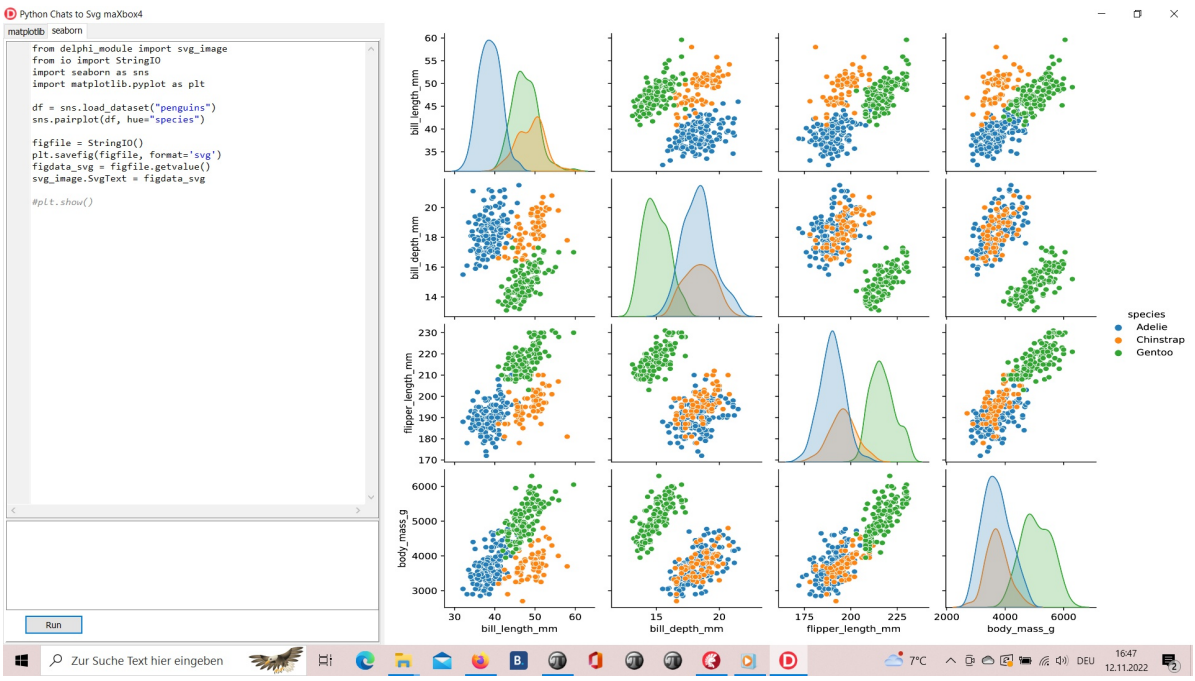
1096_2022-11-12_svg_python2seaborn.png

```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 344 entries, 0 to 343
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   species            344 non-null    object
 1   island             344 non-null    object
 2   culmen_length_mm   342 non-null    float64
 3   culmen_depth_mm    342 non-null    float64
 4   flipper_length_mm  342 non-null    float64
 5   body_mass_g        342 non-null    float64
 6   sex                334 non-null    object
dtypes: float64(4), object(3)
memory usage: 18.9+ KB
memory shape core cube #7
```

The dataset consists of 7 columns.

- **species: penguin species (Chinstrap, Adélie, or Gentoo)**
- **culmen_length_mm**
- **culmen depth (mm)**
- **flipper_length_mm: flipper length (mm)**

- **body_mass_g: body mass (g)**
- **island: island name (Dream, Torgersen, or Biscoe) in the Palmer Archipelago (Antarctica)**
- **sex: penguin sex**



1096_2022-11-12_svg_python_delphi_scalable.png

Seaborn is a **Python** data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. On one short code snippets we get the scatter-plot above with the following lines:

```
import numpy as np
import pandas as pd
from io import StringIO
import matplotlib.pyplot as plt
import seaborn as sns
df = sns.load_dataset("penguins")
sns.pairplot(df, hue="species")
figfile = StringIO()
plt.savefig(figfile, format="svg")
plt.show()
print(df.info())
```

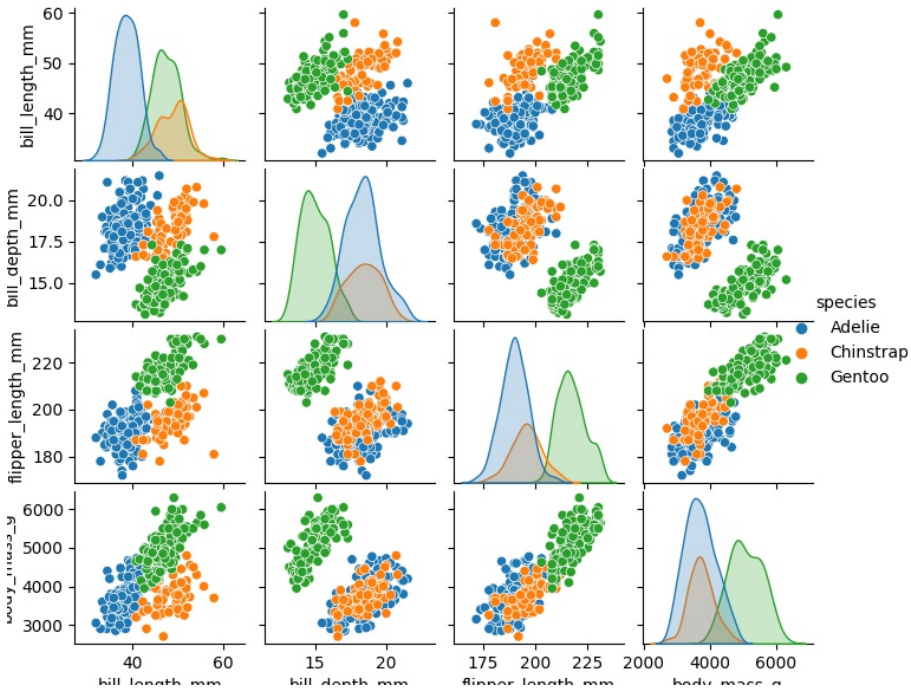
A scatter plot (aka scatter chart, scatter graph) uses dots to represent values for two different numeric variables. The position of each dot on the horizontal and vertical axis indicates values for an individual data point. Scatter plots are used to observe relationships between variables.
A common modification of the basic scatter plot is the addition of a third variable. Values of the third variable can be encoded by modifying how the points are plotted. For a third variable that indicates categorical values (like geographical region, suburbans or gender), the most common encoding is through point color.
Seaborn lets you create relational plots using the *relplot()* function. The function technically lets you create more than one scatter plot.

1.2 Covariance

"Covariance" indicates the direction of the linear relationship between variables. "Correlation" on the other hand measures both the strength and direction of the linear relationship between two variables.

Source: https://tinyurl.com/yd2pezss

Second correlations describe the mutual relationship between two variables. They provide the possibility to measure the relation between any kind of data - continuous and continuous, categorical and categorical, and also continuous and categorical at the same time.



1096_Figure_1_species.png

The scatter plot which shows us the correlation with respect to other features. This method helps just to figure out the important features which account the most for the classification in our model.

1.3 Scatter Plot

A scatter plot can also be a pair plot using values for two different numeric variables. The position of each dot on the different horizontal and vertical axis indicates values for an individual data point. Scatter plots are mainly used to observe relationships between variables.
A pairplot on the other side plots a pairwise relationships in a dataset. The pairplot function creates a grid of Axes such that each variable in data will by shared in the y-axis across a single row and in the x-axis across a single column. That creates plots as shown before.

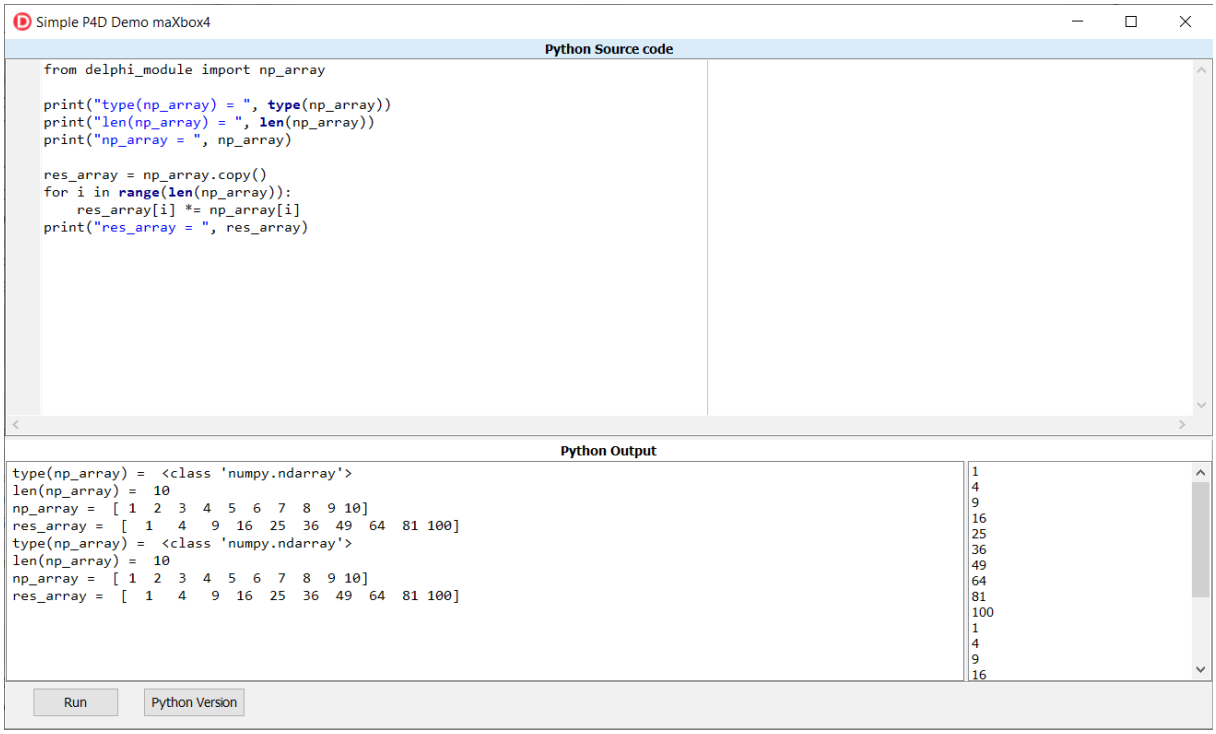
Official releases of seaborn can be installed from PyPI:

```
pip install seaborn
```

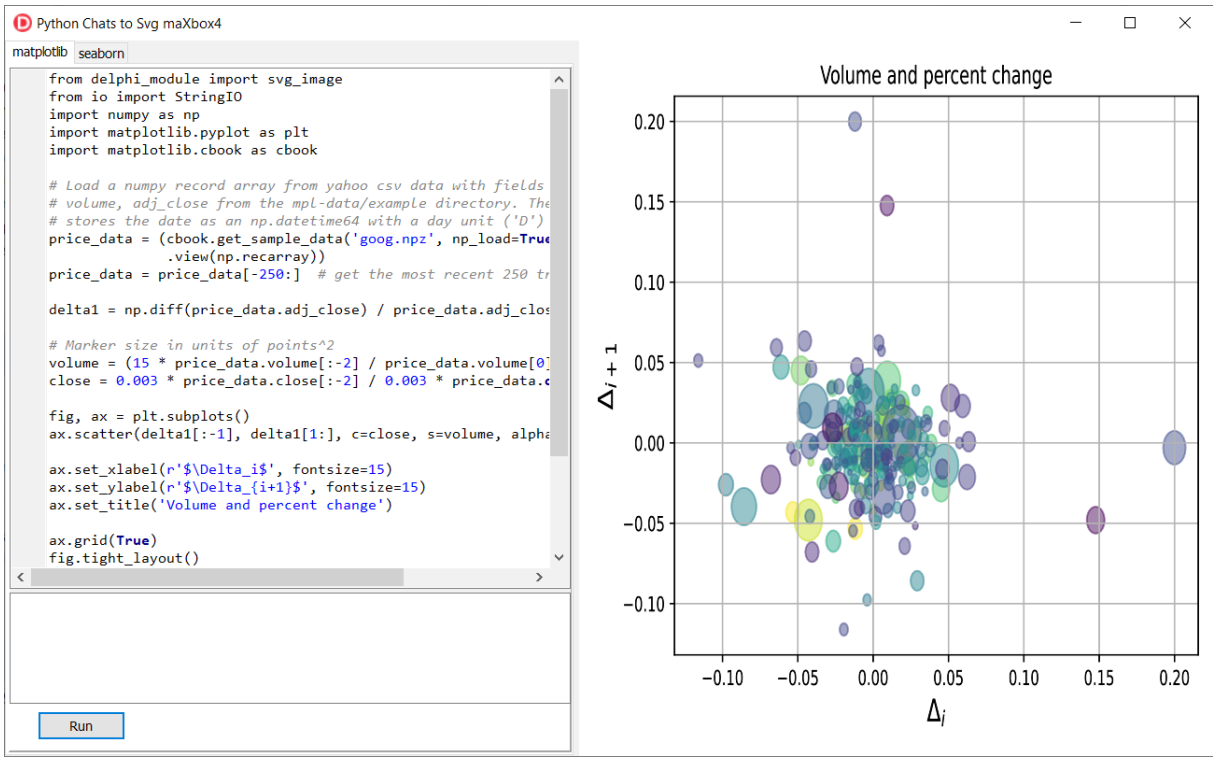
The basic invocation of `pip` will install seaborn and, if necessary, its mandatory dependencies. It is possible to include optional dependencies that give access to a few advanced features:

pip install seaborn[stats].
Numpy arrays are a good substitute for python lists. They are better than python lists. They provide faster speed and take less memory space. Let's begin with its definition for those unaware of numpy arrays. They are multi-dimensional matrices or lists of fixed size with similar elements.

Pandas is a popular Python library used to manipulate tabular data. It provides a versatile dataframe object that can read data from many popular formats, such as Excel, SQL, CSV and more.
The Pandas style API provides you with many different tools that makes working with styling tabular data much easier.



1096_2022-11-12_var_python.png

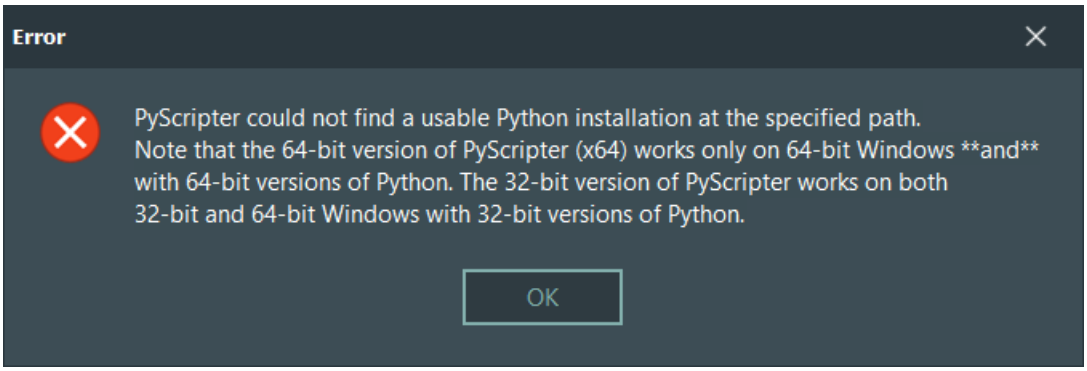


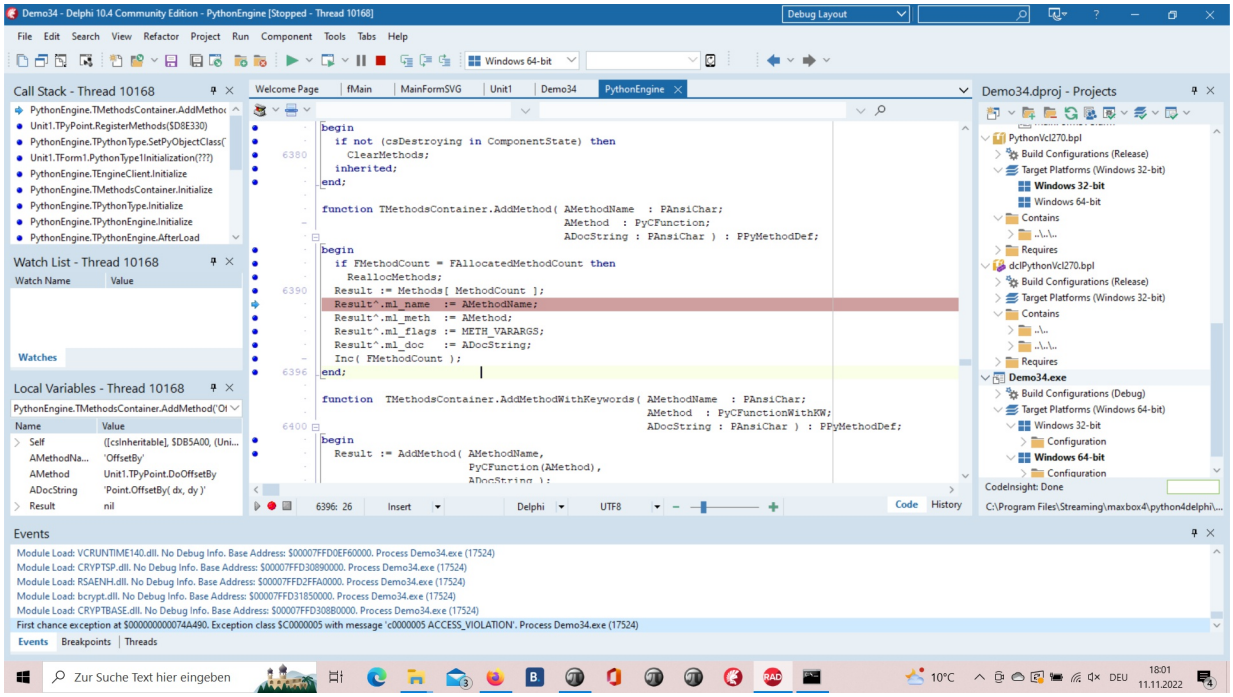
1096_2022-11-12_svg_python.png

2 Technical Description

maxkleiner (Max Kleiner) (github.com)

A Python program terminates as soon as it encounters an error. In Python, an error can be a syntax error or an exception. In his article, you will see what an exception is and how it differs from a syntax error. After that, you will learn about raising exceptions and making assertions. Then, you'll finish with a demonstration of the try and except block.
https://realpython.com/python-exceptions/



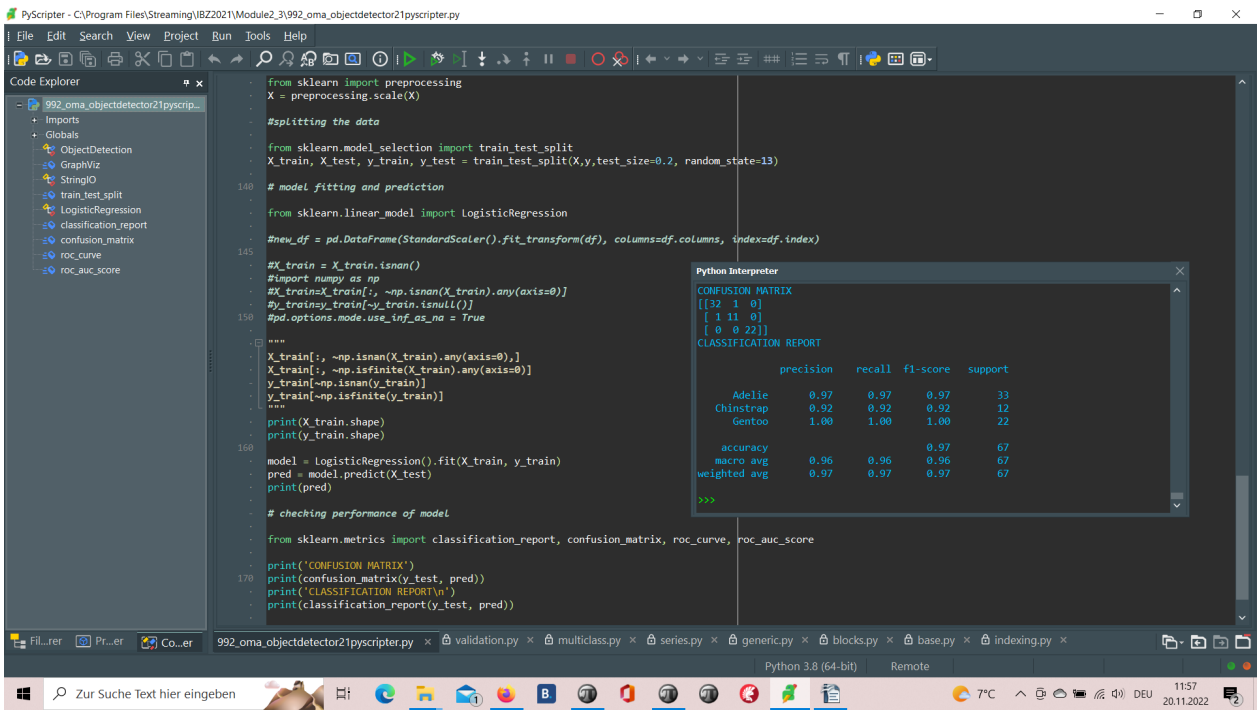


This Exception was prior to change between 32- and 64 bit.

1096_2022-11-11_exception.png

PyScripter 4.0.0 is now available at Sourceforge. PyScripter is fully programmed in Delphi. This is a major new release with an updated User Interface and many significant enhancements under the hood, that will increase the stability and improve the user experience.

Scalable Vector **Graphics** (**SVG**) is an XML-based vector image format for defining two-dimensional **graphics**, having support for interactivity and animation. The **SVG** specification is an open standard developed by the World Wide Web Consortium since 1999.
https://en.wikipedia.org/wiki/SVG



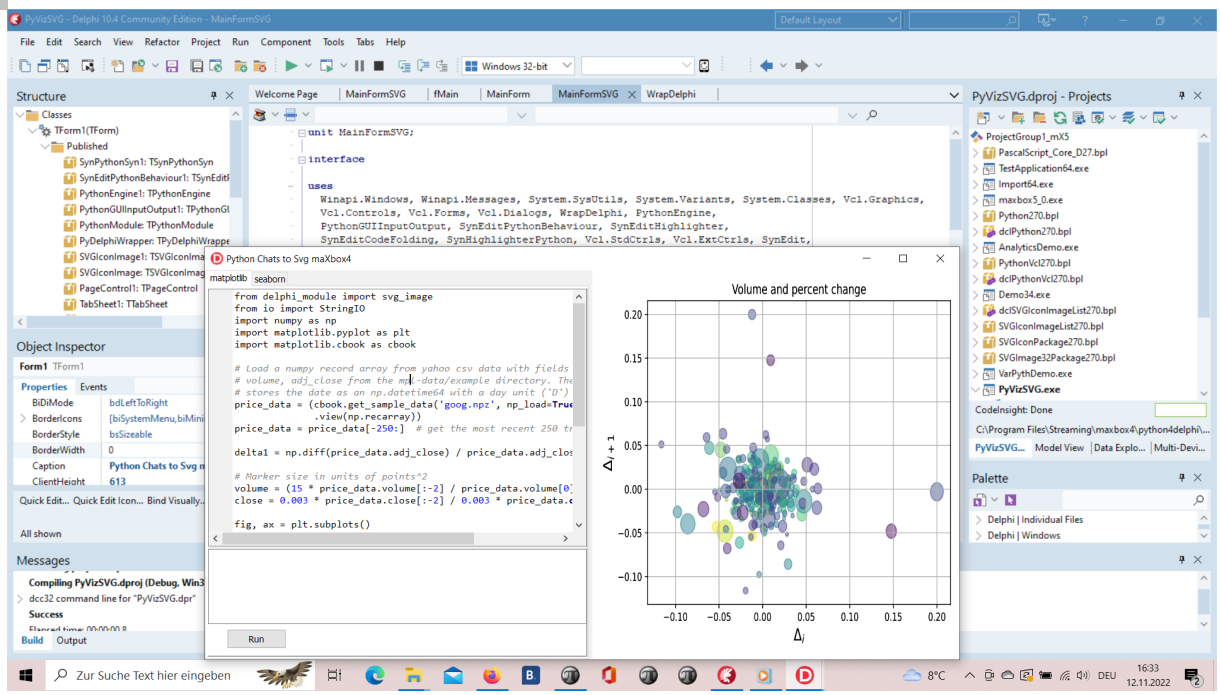
Sick of using IDLE? Want to code in a fancy new IDE? These steps will help you in downloading Python 2.7.1 or Python 3.8 and PyScripter so you can start learning and have fun with Python.

1096_2022-11-12_develop_overview2.png

https://www.instructables.com/Python-With-PyScripter/

3 Develop Overview

We value the quality of the material each student is learning. Our Instructor led classes help provide hands-on learning, one-on-one mentoring with experienced developers and peer-to-peer learning. To allow the most flexibility for students, we have Day and Night programs to choose from. *No prior coding experience required.



1096_2022-11-12_develop_overview.png

```
procedure pyBank_VCL4Python4Delphi;
var eg: TPythonEngine; sw: TStopWatch;
begin
 eg:= TPythonEngine.Create(Nil);
  try
    eg.pythonhome:= PYHOME;
    eg.loadDLL;
    println('test import
'+GetPythonEngine.EvalStr('__import__("decimal").Decimal(0.1)'));
    writeln('')
    //println(eg.EvalStr('__import__("faker").Faker()'));
    sw:= TStopWatch.Create();
    sw.Start;
    eg.execStr('import importlib.machinery, importlib.util');
    eg.execStr('from decimal import Decimal, getcontext');
    importlib.machinery.ExtensionFileLoader("DelphiVCL",'+VCLHOME+')')

    eg.execStr(ANALYTICSSVG);

//println(eg.evalStr(''#39'MSE:'#39',mean_squared_error(test_bayesian_pred,
y_test_confirmed)'));

    sw.Stop;
    //sw.ElapsedMilliseconds;
    writeln('Stop Analytics Tester1: '+sw.getValueStr)
  except
    eg.raiseError;
    writeln(ExceptionToString(ExceptionType, ExceptionParam));
  finally
    eg.Free;
    sw.Free;
    sw:= Nil;
  end;
end;
```