In [1]:
```
!which python
```

```
/anaconda3/envs/iris-course/bin/python
```

In [2]:
```
%load_ext autoreload
%autoreload 2
%matplotlib inline
```

In [5]:
```
# Standard imports
import os

# Third-party imports
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import datasets

# Local imports
```

In [4]:
```
sns.set()
```

Load data

Let's load the Iris flower dataset using scikit-learn's built-in datasets.

In [ ]:
```
datasets.load_iris?
```

Load data

Let's load the Iris flower dataset using scikit-learn's built-in datasets.

```
In [6]: datasets.load_iris?
```

```
In [ ]:
```

```
Signature: datasets.load_iris(return_X_y=False)
Docstring:
Load and return the iris dataset (classification).

The iris dataset is a classic and very easy multi-class classification
dataset.

=================   ==============
Classes                          3
Samples per class               50
Samples total                  150
Dimensionality                   4
Features            real, positive
=================   ==============

Read more in the :ref:`User Guide <iris_dataset>`.

Parameters
----------
return_X_y : boolean, default=False.
    If True, returns ``(data, target)`` instead of a Bunch object. See
    below for more information about the `data` and `target` object.
```

Let's load the Iris flower dataset using scikit-learn's built-in datasets.

```
In [6]: datasets.load_iris?
```

```
In [ ]:
```

Load and return the iris dataset (classification).

The iris dataset is a classic and very easy multi-class classification dataset.

Classes	3
Samples per class	50
Samples total	150
Dimensionality	4
Features	real, positive

Read more in the :ref:`User Guide <iris_dataset>`.

Parameters
---

return_X_y : boolean, default=False.
    If True, returns ``(data, target)`` instead of a Bunch object. See
    below for more information about the `data` and `target` object.

    .. versionadded:: 0.18

Returns

Let's load the Iris flower dataset using scikit-learn's built-in datasets.

```
In [6]: datasets.load_iris?
```

```
In [ ]:
```

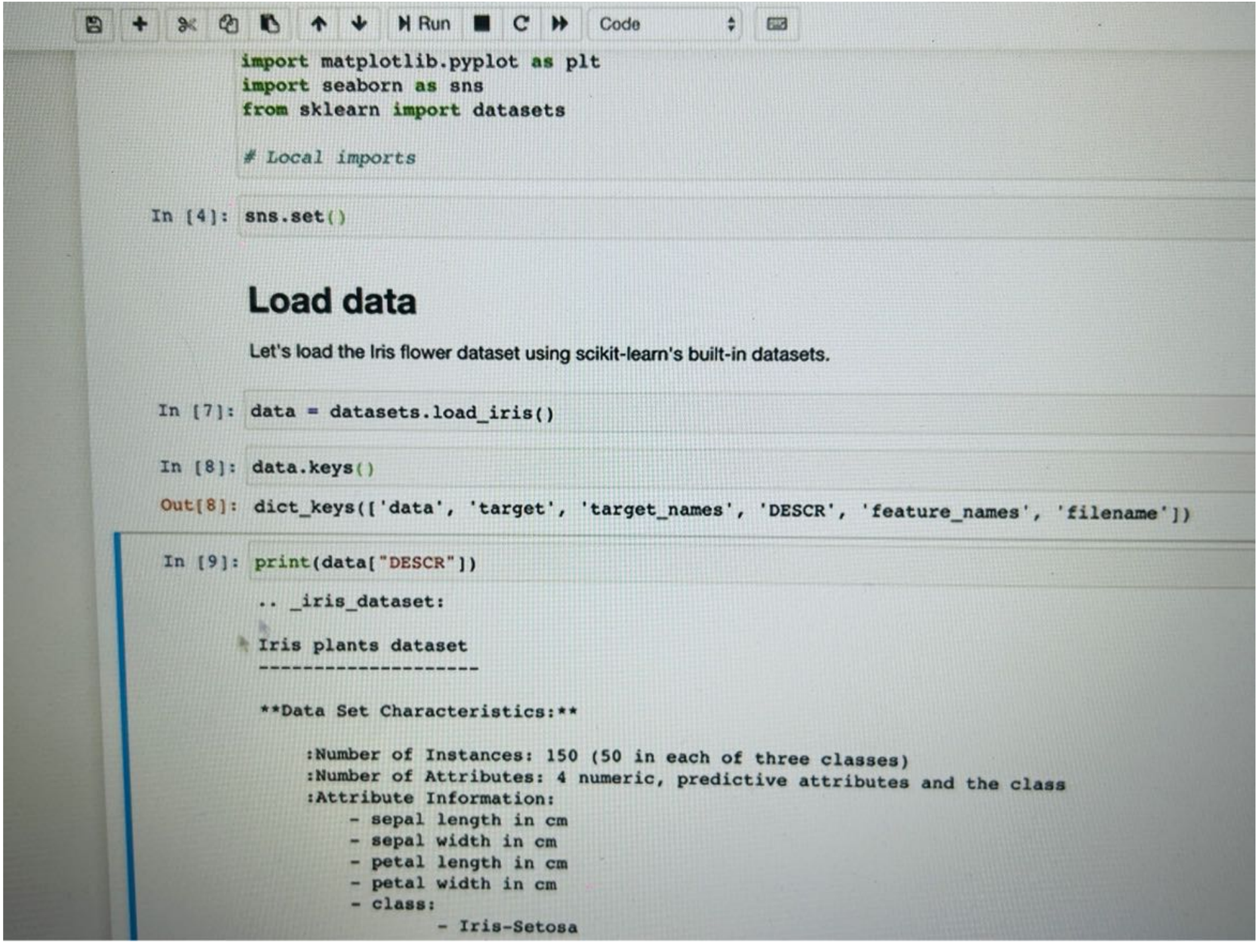
```
---------
return_X_y : boolean, default=False.
    If True, returns ``(data, target)`` instead of a Bunch object. See
    below for more information about the `data` and `target` object.

    .. versionadded:: 0.18

Returns
-------
data : Bunch
    Dictionary-like object, the interesting attributes are:
    'data', the data to learn, 'target', the classification labels,
    'target_names', the meaning of the labels, 'feature_names', the
    meaning of the features, 'DESCR', the full description of
    the dataset, 'filename', the physical location of
    iris csv dataset (added in version `0.20`).

(data, target) : tuple if ``return_X_y`` is True

    .. versionadded:: 0.18

Notes
-----
    .. versionchanged:: 0.20
```

Let's load the Iris flower dataset using scikit-learn's built-in datasets.

```
In [6]: datasets.load_iris?
```

```
In [ ]:
```

```
---------
return_X_y : boolean, default=False.
    If True, returns ``(data, target)`` instead of a Bunch object. See
    below for more information about the `data` and `target` object.

    .. versionadded:: 0.18

Returns
-------
data : Bunch
    Dictionary-like object, the interesting attributes are:
    'data', the data to learn, 'target', the classification labels,
    'target_names', the meaning of the labels, 'feature_names', the
    meaning of the features, 'DESCR', the full description of
    the dataset, 'filename', the physical location of
    iris csv dataset (added in version `0.20`).

(data, target) : tuple if ``return_X_y`` is True

    .. versionadded:: 0.18

Notes
-----
    .. versionchanged:: 0.20
```


Run
Code
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import datasets
# Local imports
In [4]: sns.set()
Load data
Let's load the Iris flower dataset using scikit-learn's built-in datasets.
In [7]: data = datasets.load_iris()
In [8]: data.keys()
Out[8]: dict_keys(['data', 'target', 'target_names', 'DESCR', 'feature_names', 'filename'])
In [9]: print(data["DESCR"])
.. _iris_dataset:
Iris plants dataset
Data Set Characteristics:
:Number of Instances: 150 (50 in each of three classes)
:Number of Attributes: 4 numeric, predictive attributes and the class
:Attribute Information:
- sepal length in cm
- sepal width in cm
- petal length in cm
- petal width in cm
- class:
- Iris-Setosa

Iris plants dataset

Data Set Characteristics:

:Number of Instances: 150 (50 in each of three classes)
:Number of Attributes: 4 numeric, predictive attributes and the class
:Attribute Information:
- sepal length in cm
- sepal width in cm
- petal length in cm
- petal width in cm
- class:
  - Iris-Setosa
  - Iris-Versicolour
  - Iris-Virginica

:Summary Statistics:

	Min	Max	Mean	SD	Class Correlation	
sepal length:	4.3	7.9	5.84	0.83	0.7826	
sepal width:	2.0	4.4	3.05	0.43	-0.4194	
petal length:	1.0	6.9	3.76	1.76	0.9490	(high!)
petal width:	0.1	2.5	1.20	0.76	0.9565	(high!)

:Missing Attribute Values: None
:Class Distribution: 33.3% for each of 3 classes.
:Creator: R.A. Fisher
:Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
:Date: July, 1988

The famous Iris database, first used by Sir R.A. Fisher. The dataset is taken from Fisher's paper. Note that it's the same as in R, but not as in the UCI Machine Learning Repository, which has two wrong data points.

- Iris-Versicolour
- Iris-Virginica

:Summary Statistics:

	Min	Max	Mean	SD	Class Correlation	
sepal length:	4.3	7.9	5.84	0.83	0.7826	
sepal width:	2.0	4.4	3.05	0.43	-0.4194	
petal length:	1.0	6.9	3.76	1.76	0.9490	(high!)
petal width:	0.1	2.5	1.20	0.76	0.9565	(high!)

:Missing Attribute Values: None
:Class Distribution: 33.3% for each of 3 classes.
:Creator: R.A. Fisher
:Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
:Date: July, 1988

The famous Iris database, first used by Sir R.A. Fisher. The dataset is taken from Fisher's paper. Note that it's the same as in R, but not as in the UCI Machine Learning Repository, which has two wrong data points.

This is perhaps the best known database to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is referenced frequently to this day. (See Duda & Hart, for example.) The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other.

.. topic:: References

- Fisher, R.A. "The use of multiple measurements in taxonomic problems" Annual Eugenics, 7, Part II, 179-188 (1936); also in "Contributions to Mathematical Statistics" (John Wiley, NY, 1950).
- Duda, R.O., & Hart, P.E. (1973) Pattern Classification and Scene Analysis. (Q327.D83) John Wiley & Sons. ISBN 0-471-22361-1. See page 218.
- Dasarathy, B.V. (1980) "Nosing Around the Neighborhood: A New System

```
petal length:    1.0   6.9    3.76    1.76     0.9490  (high!)
petal width:     0.1   2.5    1.20    0.76     0.9565  (high!)
================ ==== ==== ======= ===== ====================

:Missing Attribute Values: None
:Class Distribution: 33.3% for each of 3 classes.
:Creator: R.A. Fisher
:Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
:Date: July, 1988

The famous Iris database, first used by Sir R.A. Fisher. The dataset is taken
from Fisher's paper. Note that it's the same as in R, but not as in the UCI
Machine Learning Repository, which has two wrong data points.

This is perhaps the best known database to be found in the
pattern recognition literature.  Fisher's paper is a classic in the field and
is referenced frequently to this day.  (See Duda & Hart, for example.)  The
data set contains 3 classes of 50 instances each, where each class refers to a
type of iris plant.  One class is linearly separable from the other 2; the
latter are NOT linearly separable from each other.

.. topic:: References

   - Fisher, R.A. "The use of multiple measurements in taxonomic problems"
     Annual Eugenics, 7, Part II, 179-188 (1936); also in "Contributions to
     Mathematical Statistics" (John Wiley, NY, 1950).
   - Duda, R.O., & Hart, P.E. (1973) Pattern Classification and Scene Analysis.
     (Q327.D83) John Wiley & Sons.  ISBN 0-471-22361-1.  See page 218.
   - Dasarathy, B.V. (1980) "Nosing Around the Neighborhood: A New System
     Structure and Classification Rule for Recognition in Partially Exposed
     Environments".  IEEE Transactions on Pattern Analysis and Machine
     Intelligence, Vol. PAMI-2, No. 1, 67-71.
   - Gates, G.W. (1972) "The Reduced Nearest Neighbor Rule".  IEEE Transactions
     on Information Theory, May 1972, 431-433.
   - See also: 1988 MLC Proceedings, 54-64.  Cheeseman et al"s AUTOCLASS II
     conceptual clustering system finds 3 classes in the data.
   - Many, many more ...
```

In [ ]: