https://jakevdp.github.io/PythonDataScienceHandbook/05.01-what-is-machine-learning.html

Machine Learning

Building models of data

Machine learning

Building mathematical models to help understand data

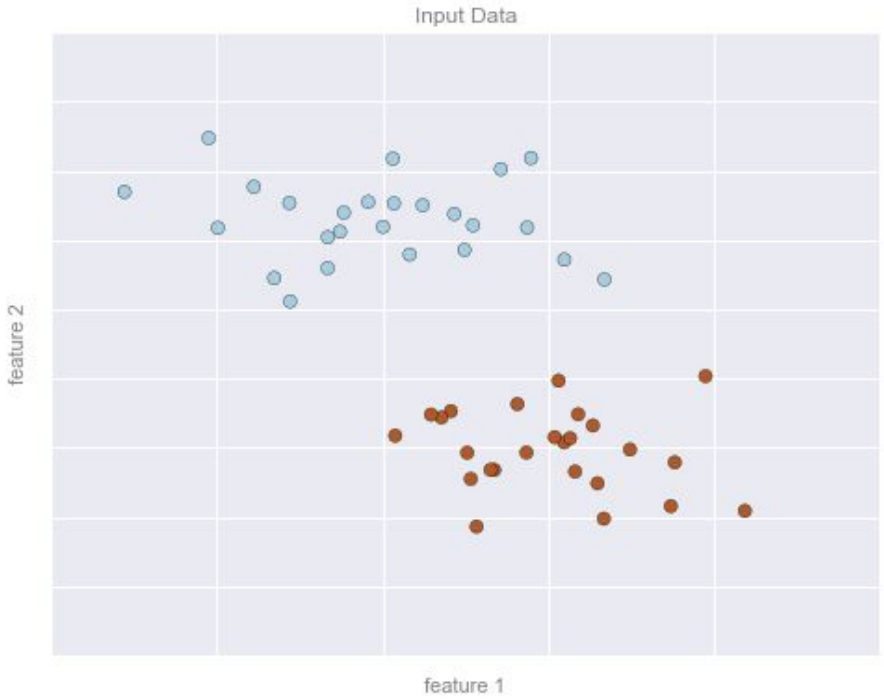
“Learning” - these models have *tunable* parameters that can be adapted to observed data

- Requirement of Machine Learning - Lotsa data
  - Training data - to fit a model
  - Observational data - to test predictions

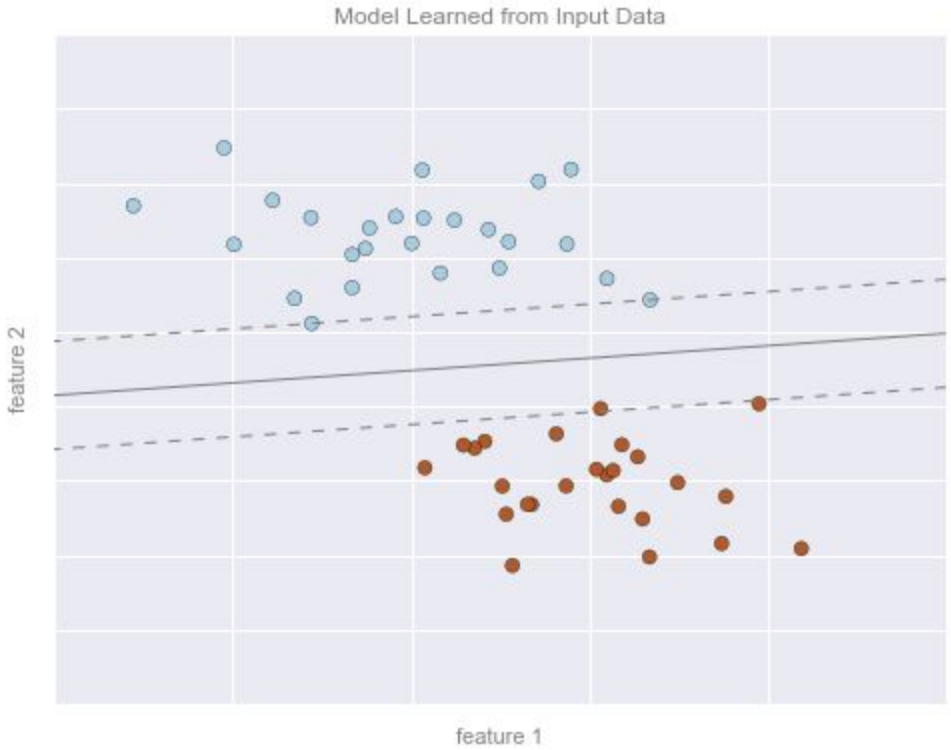
Supervised vs Unsupervised learning

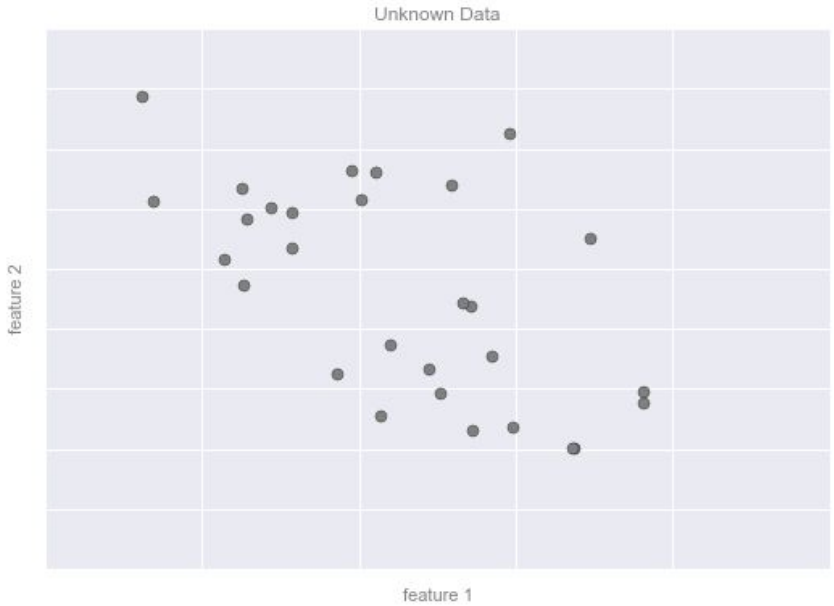
- Supervised learning involves modeling relationship between measured features of data (or “labeled data”)
  - Classification
  - Regression
- Unsupervised learning involves modeling features of a dataset without reference to any label
  - “Letting data speak for itself”
  - Clustering - distinct groups of data
  - Dimensionality reduction - succinct representations of the data
- Semi-supervised learning

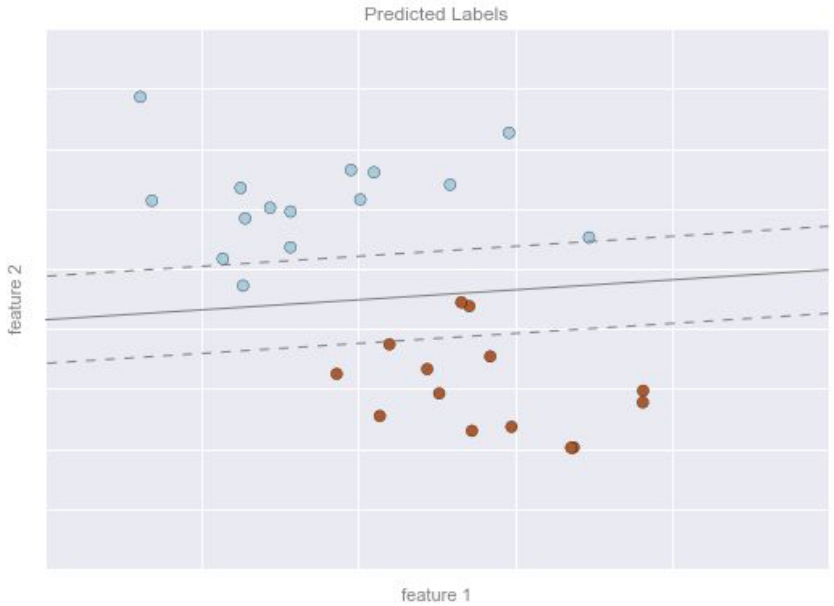
Supervised Learning - Classification



Visual representation of a trained model




Unknown Data
feature 2
feature 1


Predicted Labels
feature 2
feature 1

Some classification algorithms

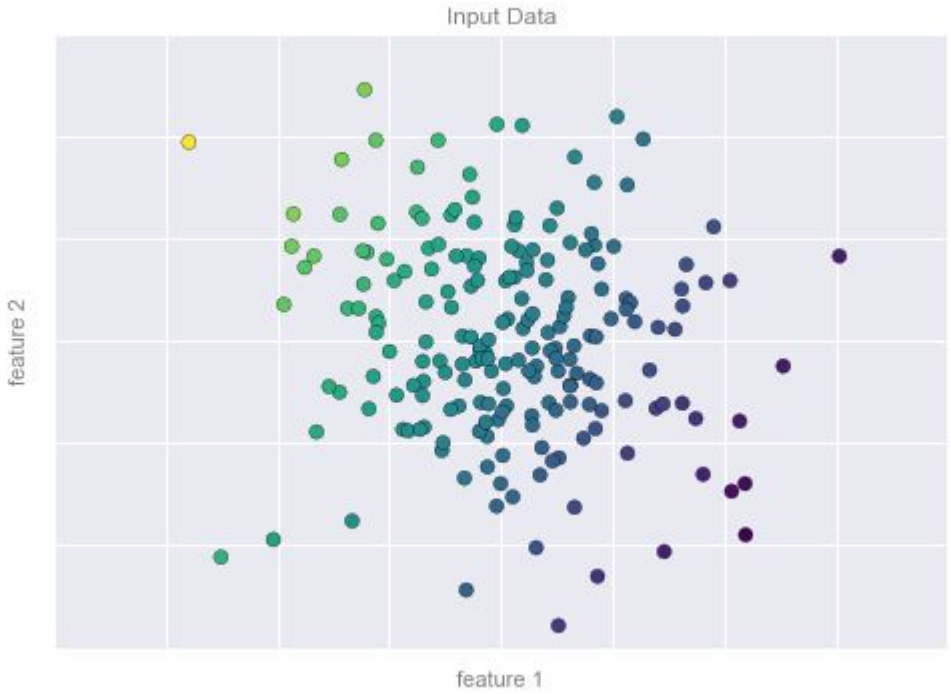
Naive Bayes Classification

Support Vector Machines

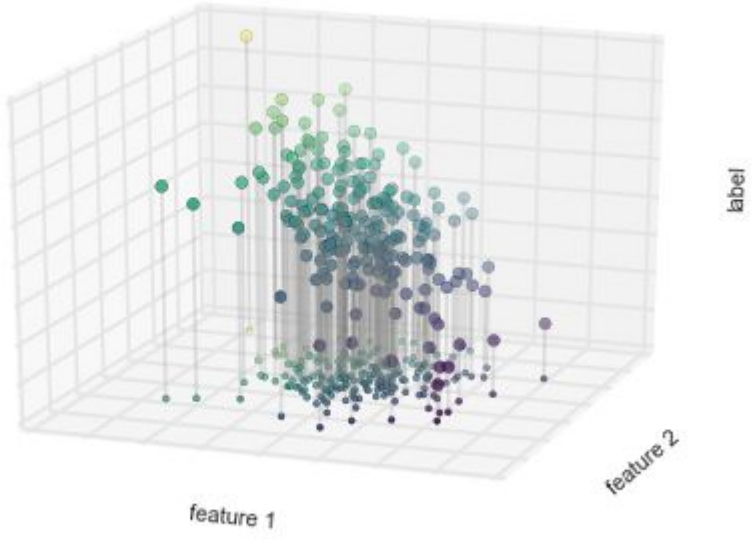
Random Forest Classification

Note that classification works with DISCRETE labels. What do we do with CONTINUOUS QUANTITIES?

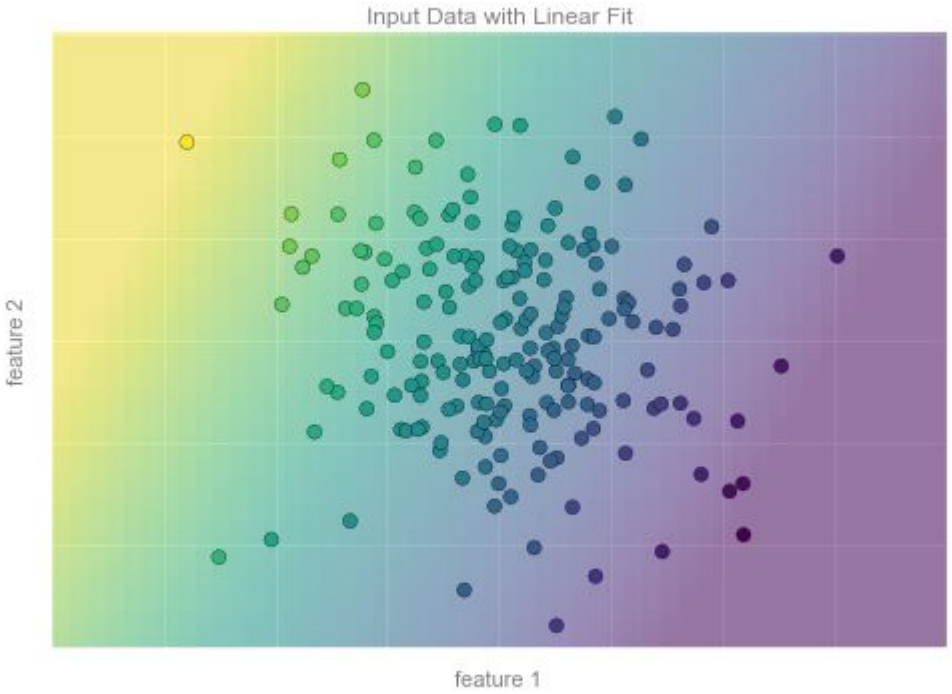
Continuous labels



Visualization of treating label as third dimension



Plane of fit to predict labels



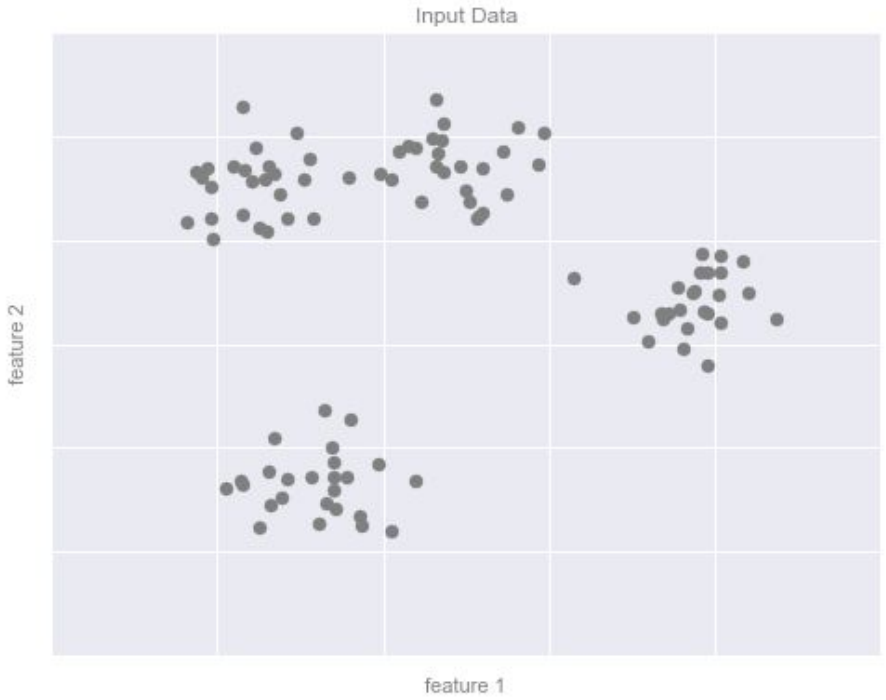
Regression algorithms

Linear Regression

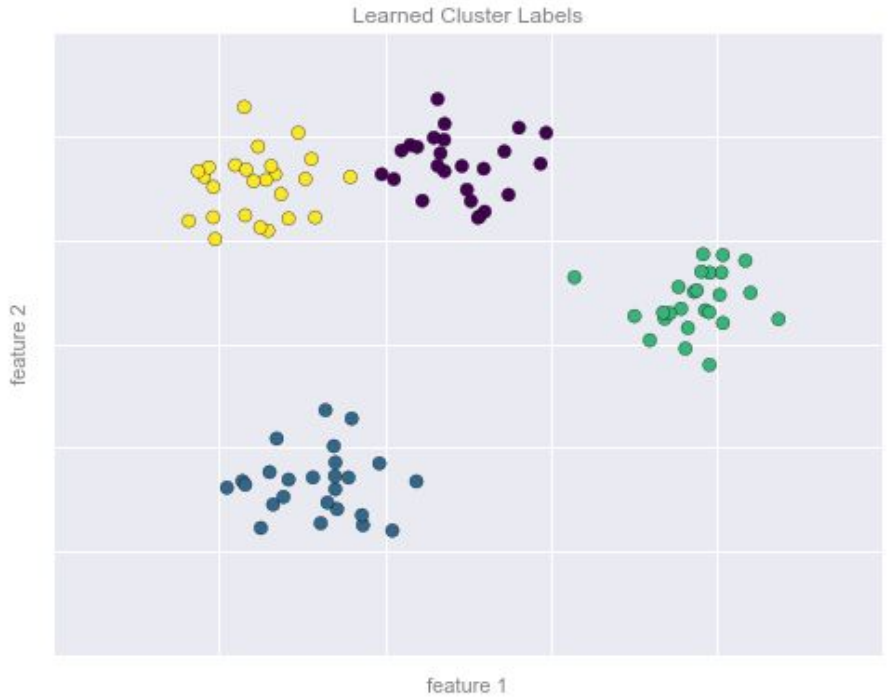
Support Vector Machines

Random Forest Regression

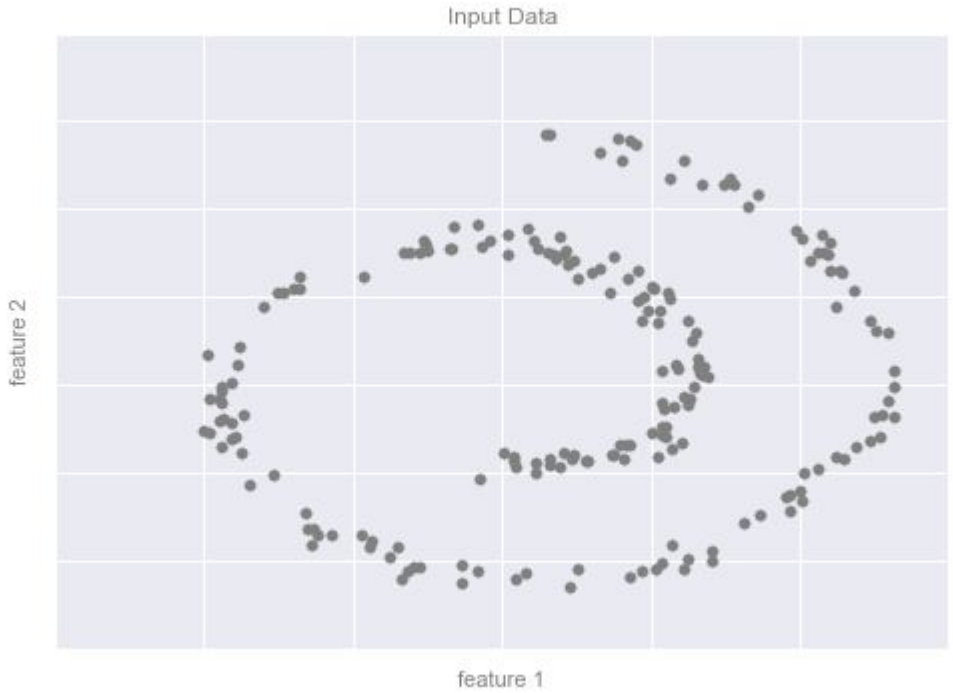
Unsupervised Clustering:



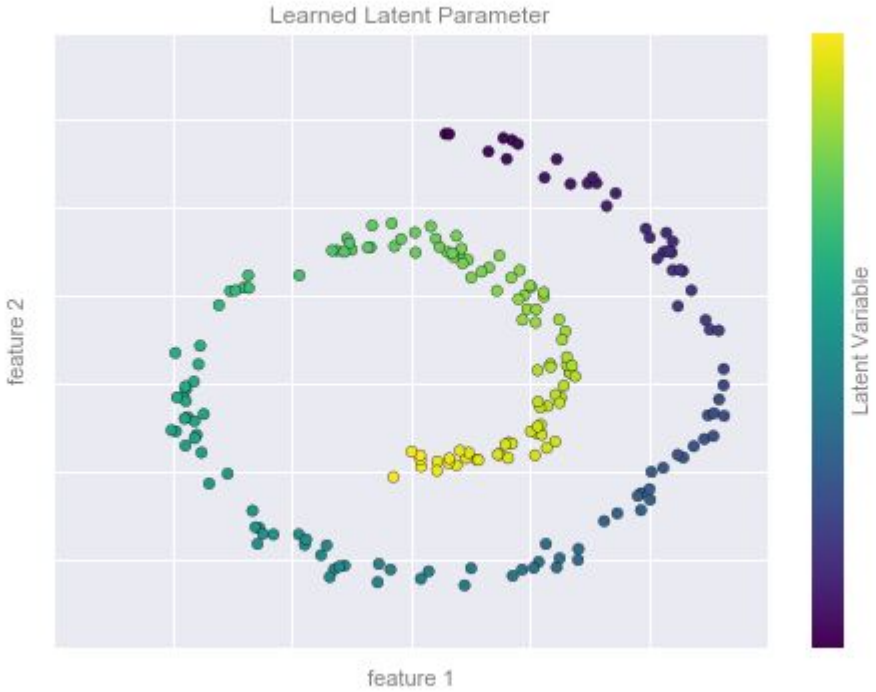
K-means clustering



Unsupervised: Dimensionality Reduction



Dimensionality reduction



Scikit-learn: Python package for machine learning

- Scikit-learn tutorials are based on standard datasets and practices
- A basic table is a two-dimensional grid of data, in which the rows represent individual elements of the dataset, and the columns represent quantities related to each of these elements

```
import seaborn as sns
iris = sns.load_dataset('iris')
iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

Y: Target array or label

Target array: usually one dimensional, with length `n_samples`, and is generally contained in a NumPy array or Pandas `Series`.

The target array may have continuous numerical values, or discrete classes/labels. While some Scikit-Learn estimators do handle multiple target values in the form of a two-dimensional, `[n_samples, n_targets]` target array, we will primarily be working with the common case of a one-dimensional target array.

Target array = what we are predicting

The distinguishing feature of the target array is that it is usually the quantity we want to *predict from the data*: in statistical terms, it is the dependent variable. For example, in the preceding data we may wish to construct a model that can predict the species of flower based on the other measurements; in this case, the `species` column would be considered the target array.