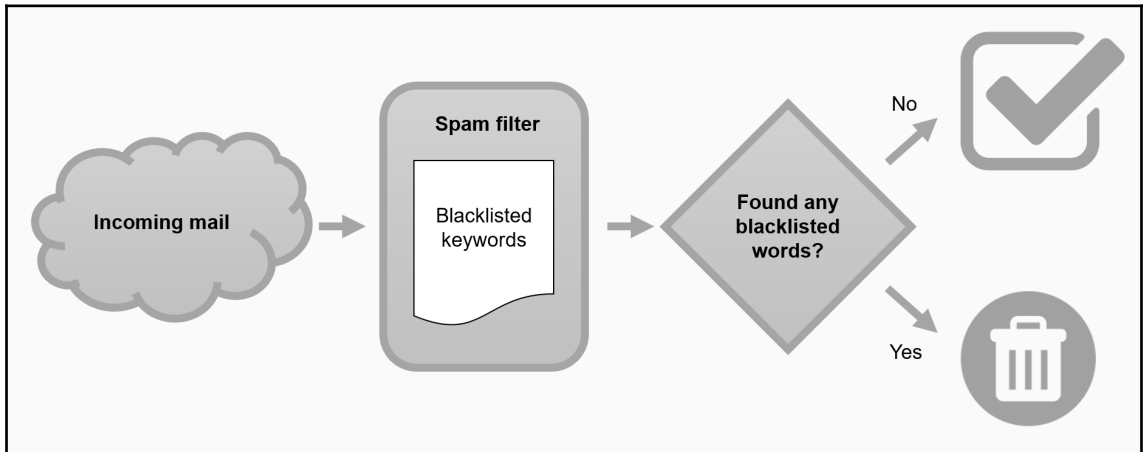
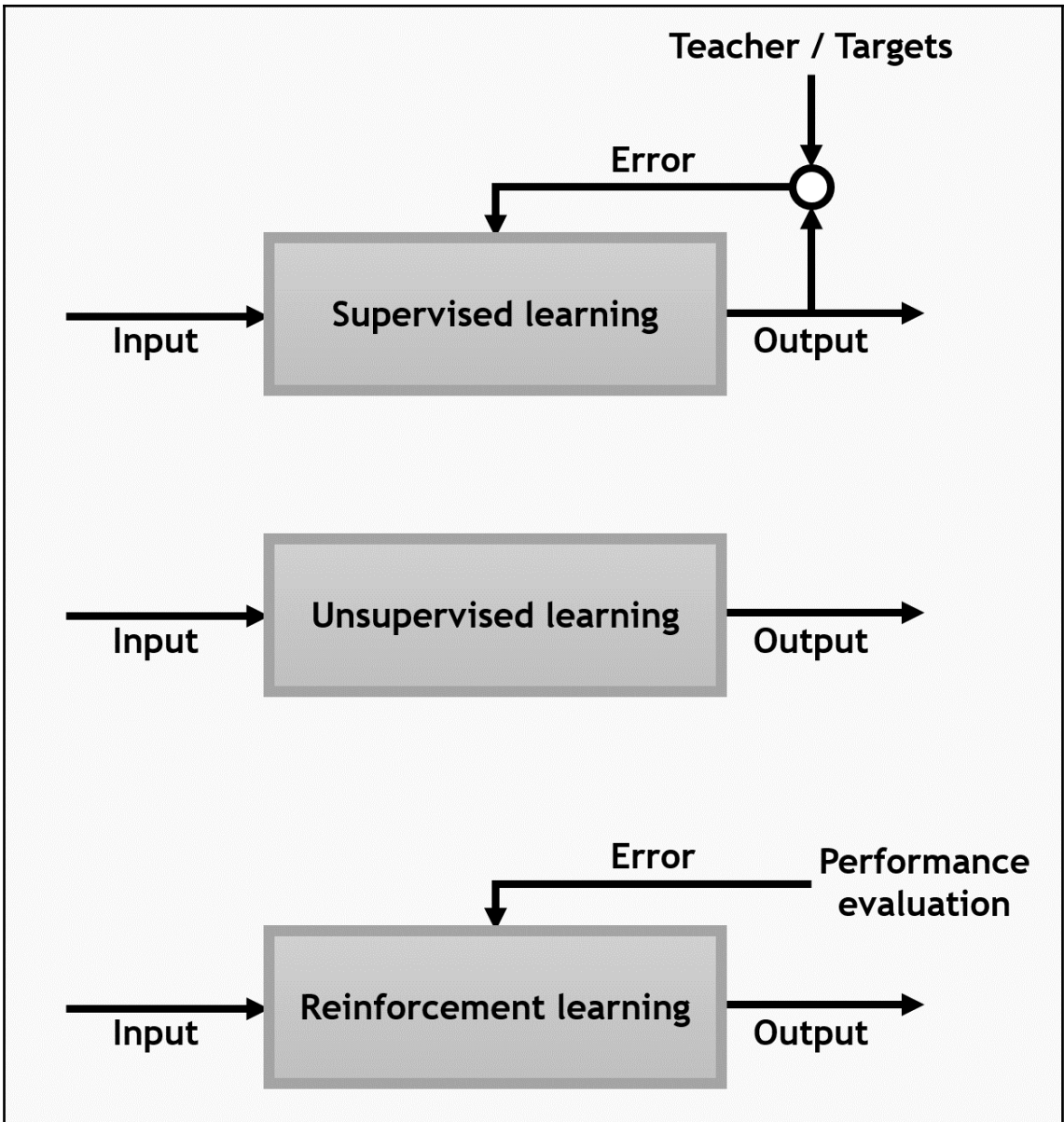


Chapter 1: A Taste of Machine Learning



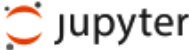


```

C:\Users\mbeyeler>anaconda search -t conda opencv
Using Anaconda API: https://api.anaconda.org
Run 'anaconda show <USER/PACKAGE>' to get more details:
Packages:

```

Name	Version	Package Types	Platforms
???/opencv	2.4.7	conda	win-64
: http://opencv.org/			
Changxu/opencv3	3.1.0_dev	conda	linux-64
Definiter/opencv	2.4.12	conda	linux-64
FlyEM/opencv	2.4.10.1	conda	linux-64, osx-64
: Open source computer vision C++ library			
JaimeIvanCervantes/opencv	2.4.9.99	conda	linux-64
RahulJain/opencv	2.4.12	conda	linux-64, win-64, osx-64
anaconda-backup/opencv	3.1.0	conda	linux-64, linux-32, osx-64
anaconda/opencv	3.1.0	conda	linux-64, linux-32, osx-64
andywocky/opencv	2.4.9	conda	osx-64
asmeurer/opencv	2.4.9	conda	osx-64
bgreen-litl/opencv	2.4.9	conda	osx-64
bwsprague/opencv	2.4.9.1	conda	osx-64
cfobel/opencv-helpers	0.1.post1	conda	win-32
clg_boar/opencv3	3.0.0	conda	linux-64, win-64
clinicalgraphics/opencv		conda	linux-64, win-32, win-64, linux-32, osx-64
conda-forge/opencv	3.1.0	conda	linux-64, win-32, win-64, osx-64
: Computer vision and machine learning software library.			



Quit Logout

Files Running Clusters

Select items to perform actions on them.
Upload New ▾ ↻

	0 ▾	/ OpenCV-ML / notebooks	Name ↓	Last Modified	File size
		..		seconds ago	
<input type="checkbox"/>		data		a day ago	
<input type="checkbox"/>		figures		a day ago	
<input type="checkbox"/>		00.00-Preface.ipynb		a day ago	9.18 kB
<input type="checkbox"/>		00.01-Foreword-by-Ariel-Rokem.ipynb		a day ago	5.55 kB
<input type="checkbox"/>		01.00-A-Taste-of-Machine-Learning.ipynb		a day ago	19.7 kB
<input type="checkbox"/>		02.00-Working-with-Data-in-OpenCV.ipynb		a day ago	7.7 kB

jupyter 01.00-A-Taste-of-Machine-Learning (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Run Code

Verifying the installation

It's a good idea to double-check your installation. While our terminal is still open, we fire up IPython, which is an interactive shell to run Python commands:

```
$ ipython
```

Now make sure that you are running Python 3.6 and not Python 2.7. You might see the version number displayed in IPython's welcome message. If not, you can run the following commands:

```
In [1]: import sys
print(sys.version)

3.6.0 | packaged by conda-forge | (default, Feb 9 2017, 14:36:55)
[GCC 4.8.2 20140120 (Red Hat 4.8.2-15)]
```

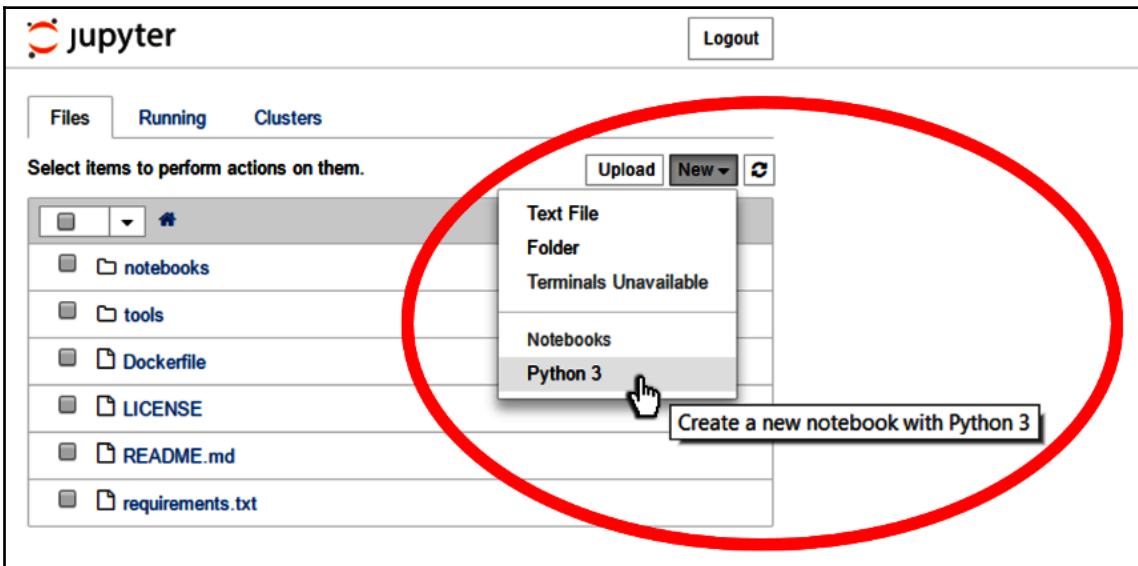
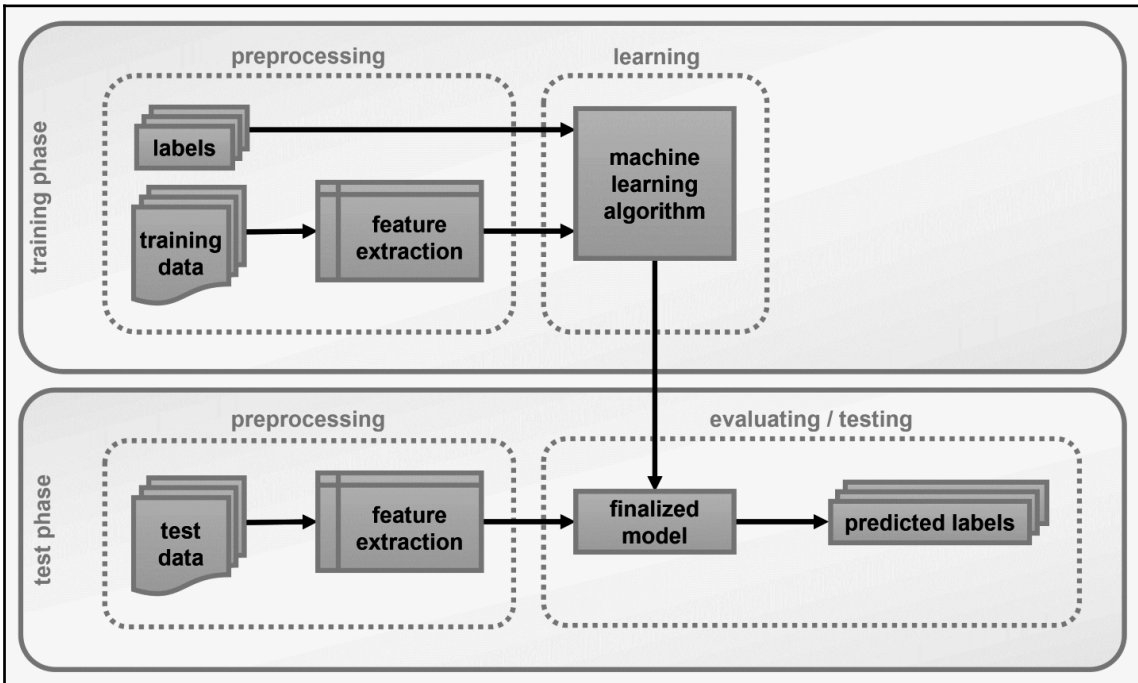
Now try to import OpenCV:



```
In [2]: import cv2
```


You should get no error messages. Then try to find out the version number:













```
In [3]: cv2.__version__
Out[3]: '4.0.0'
```

Chapter 2: Working with Data in OpenCV

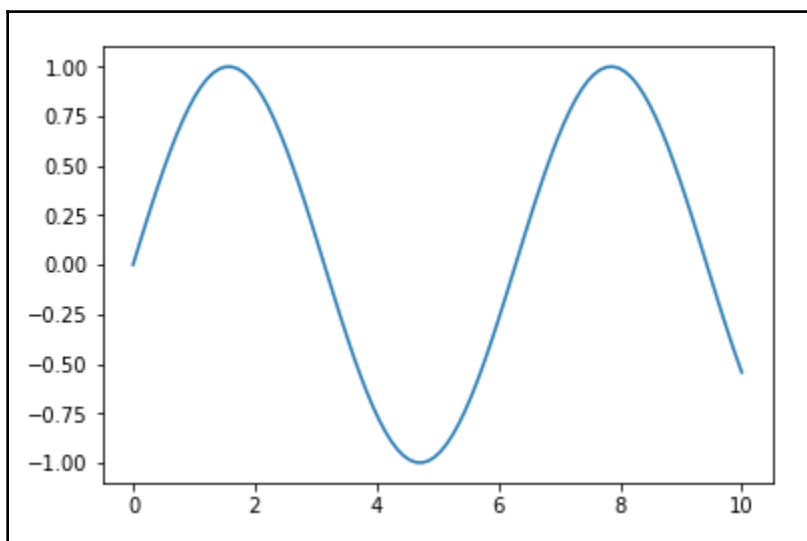


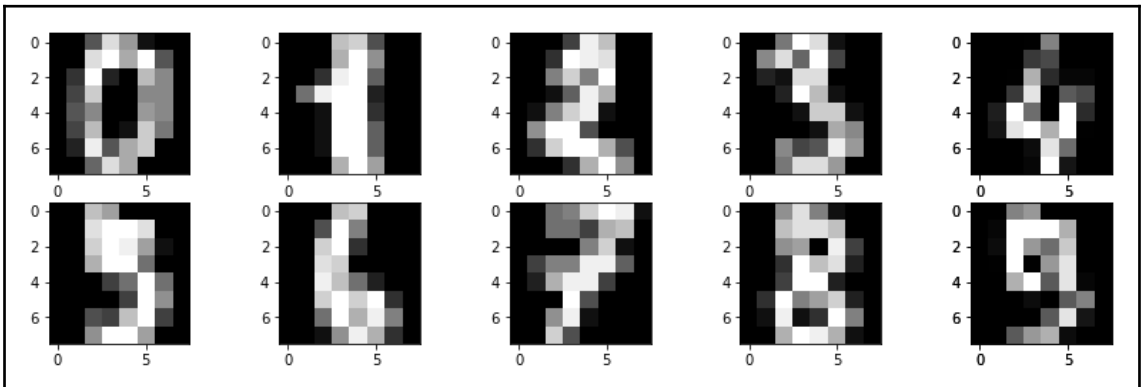
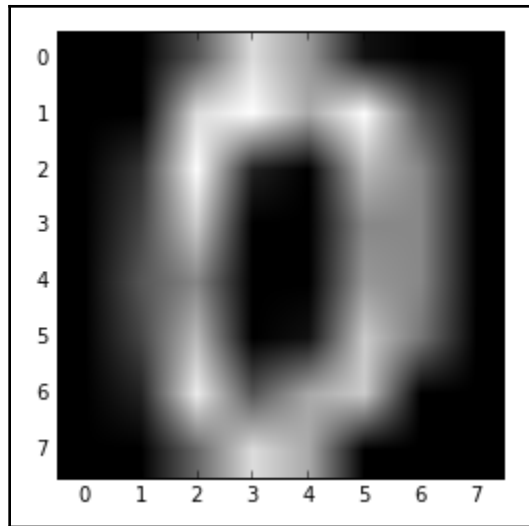
 **jupyter** Untitled Last Checkpoint: a few seconds ago (unsaved changes)  Logout

File Edit View Insert Cell Kernel Widgets Help Python 3 

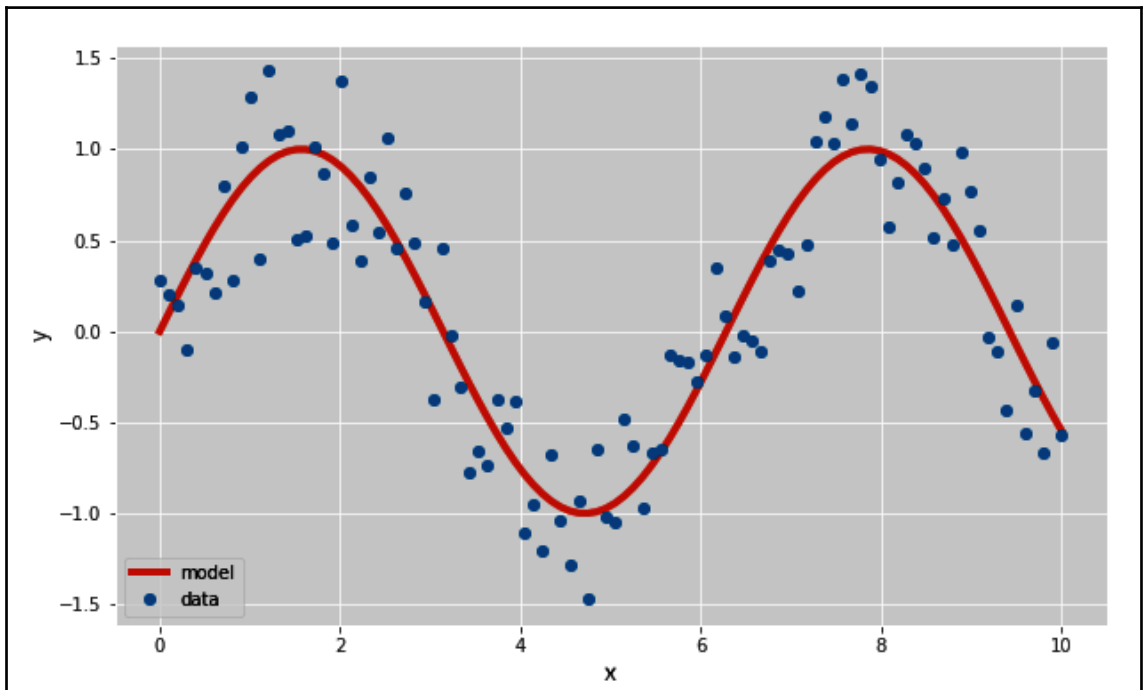
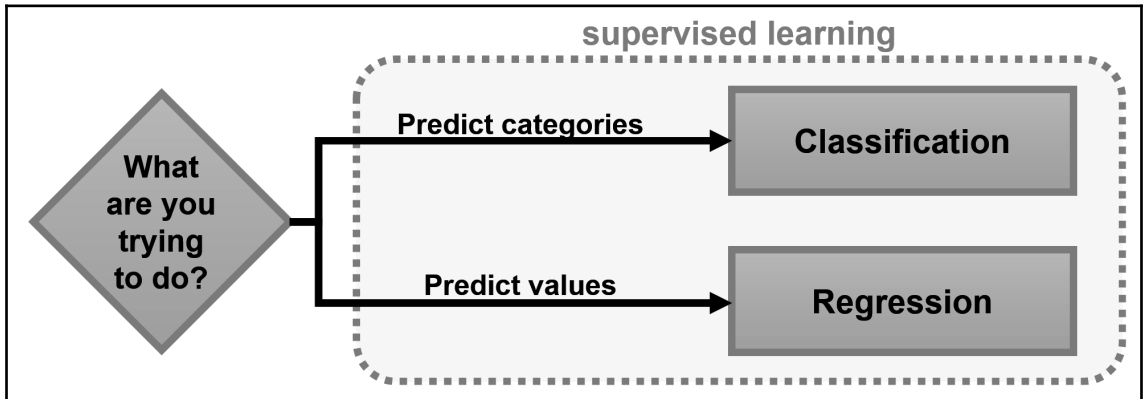
          Code   CellToolbar

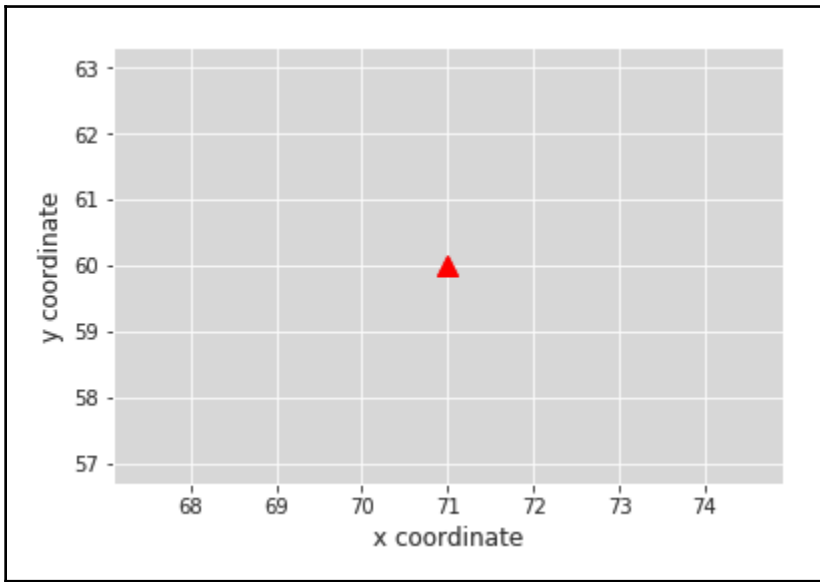
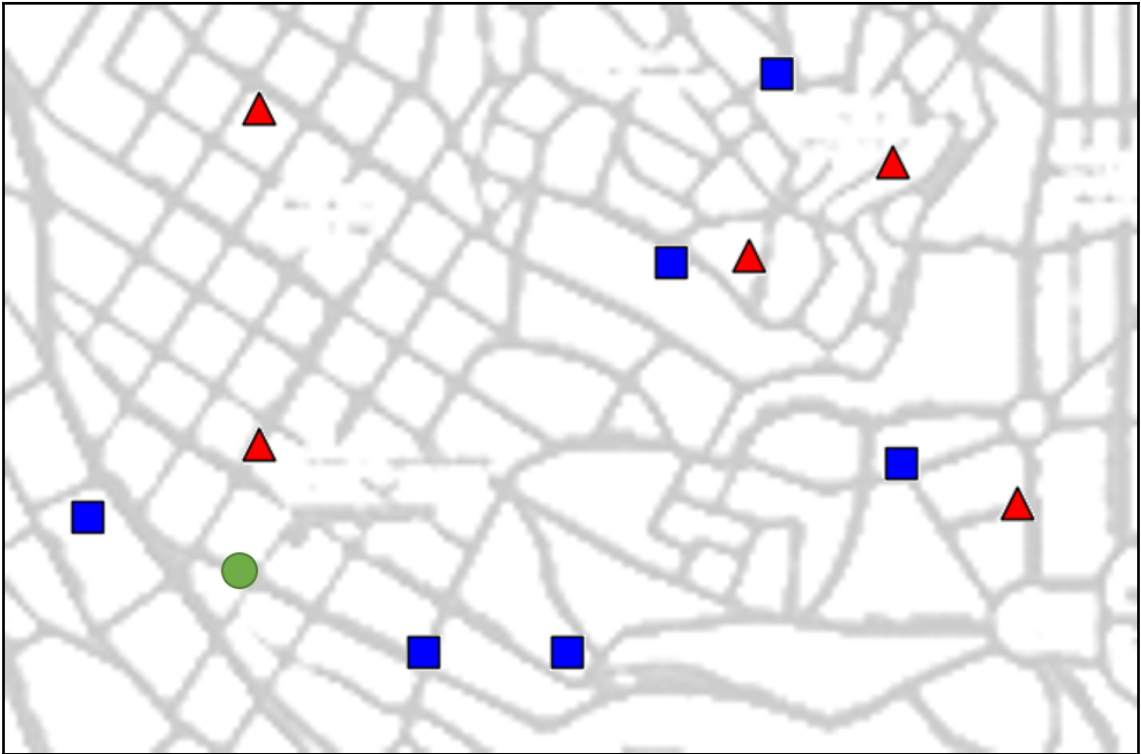
In []:

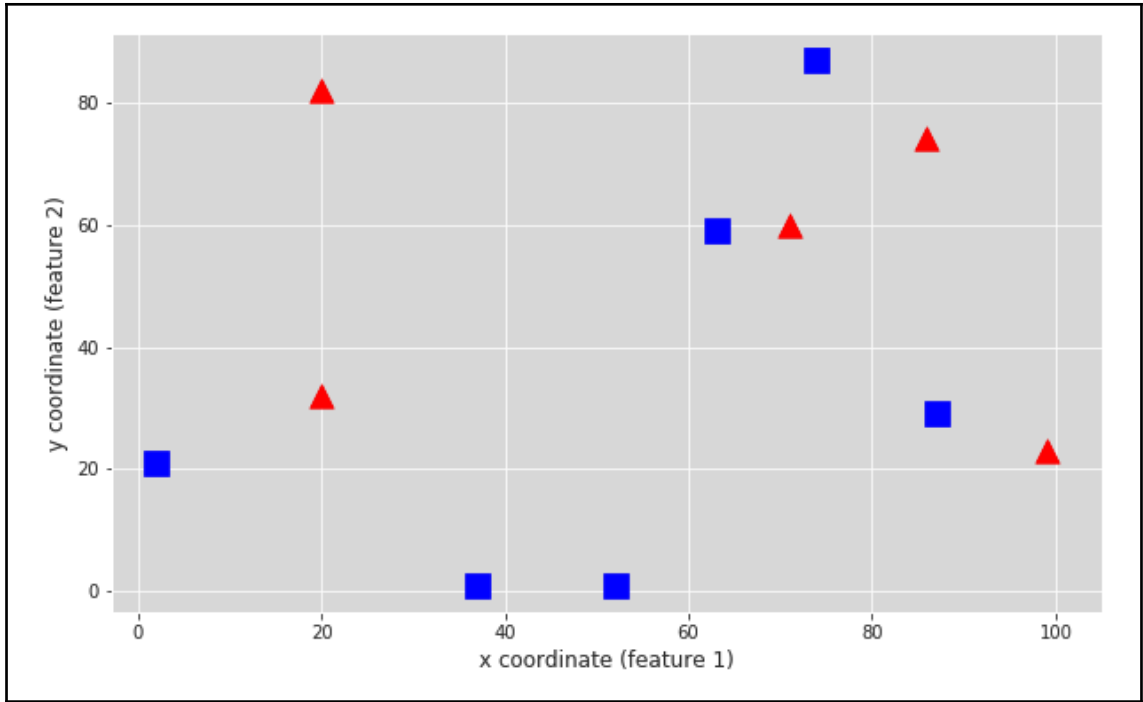


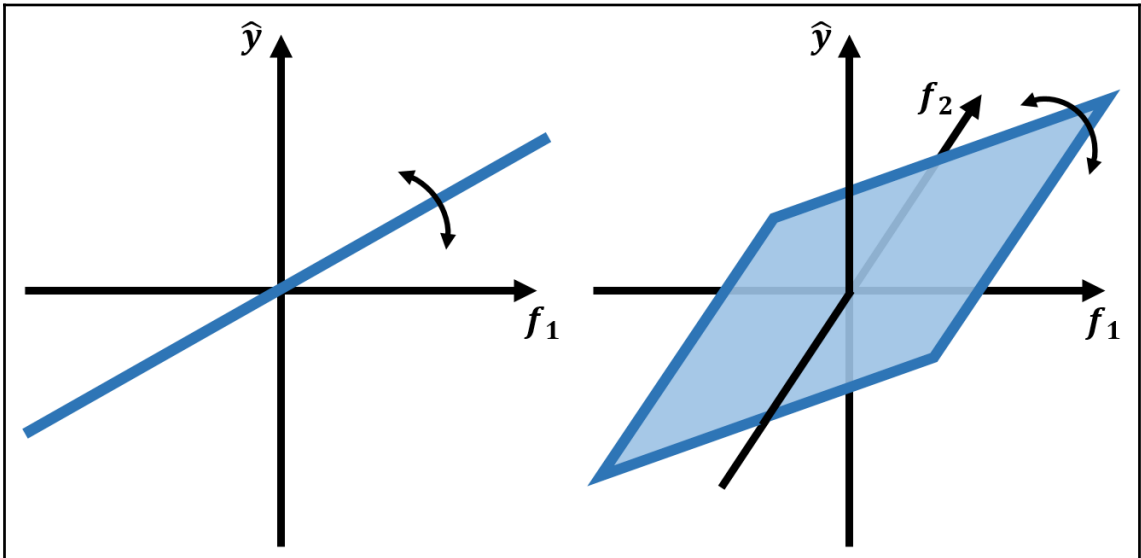
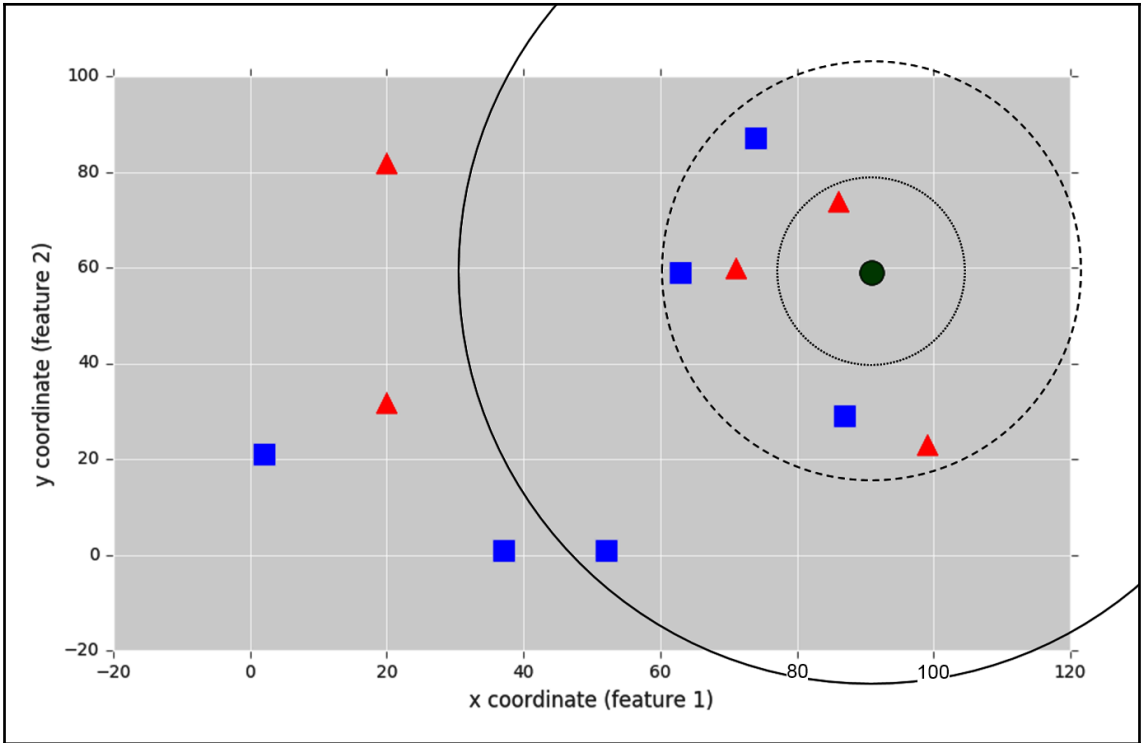


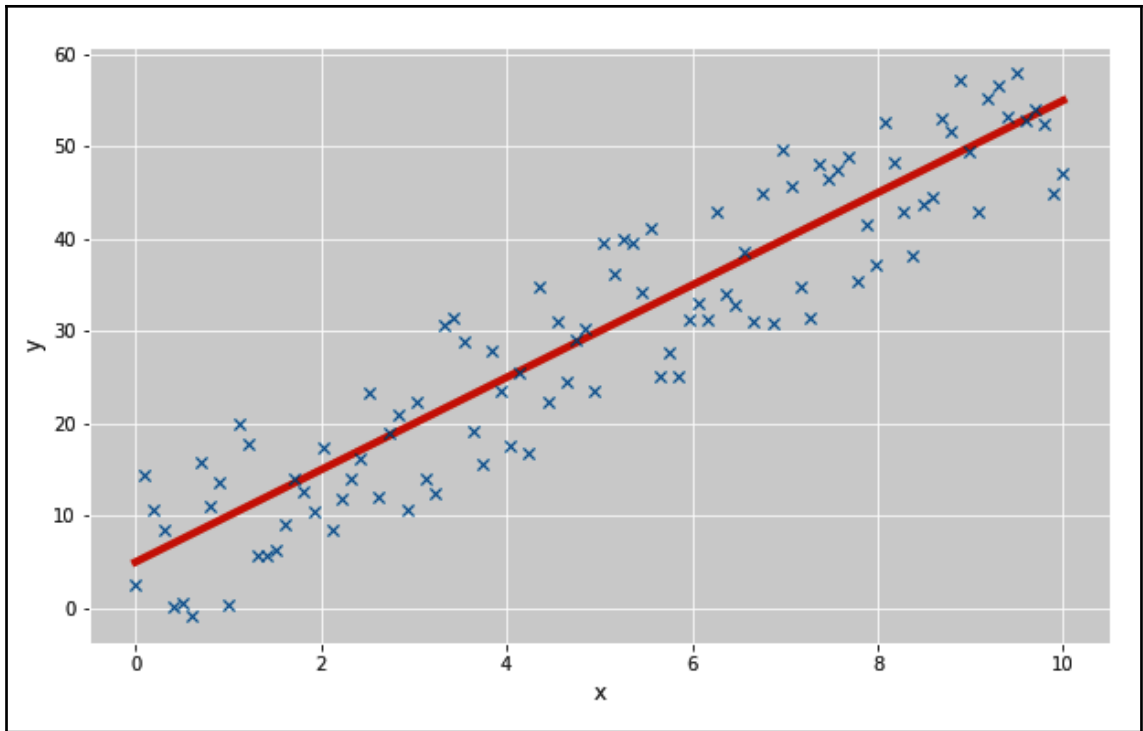
Chapter 3: First Steps in Supervised Learning

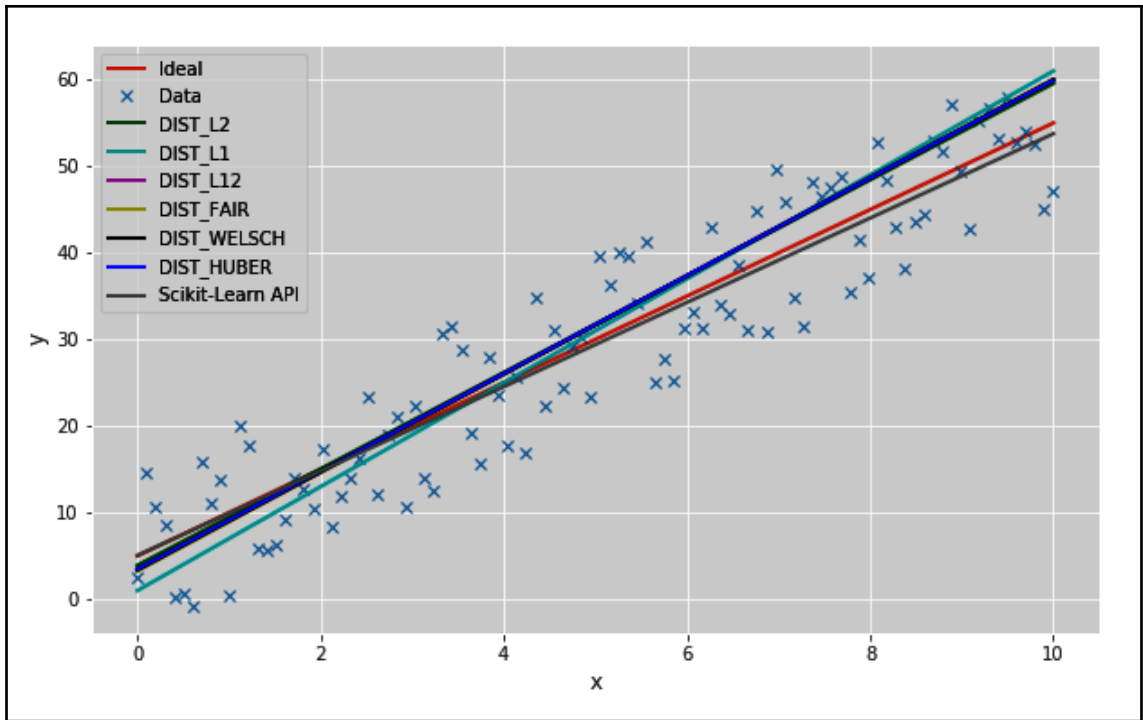


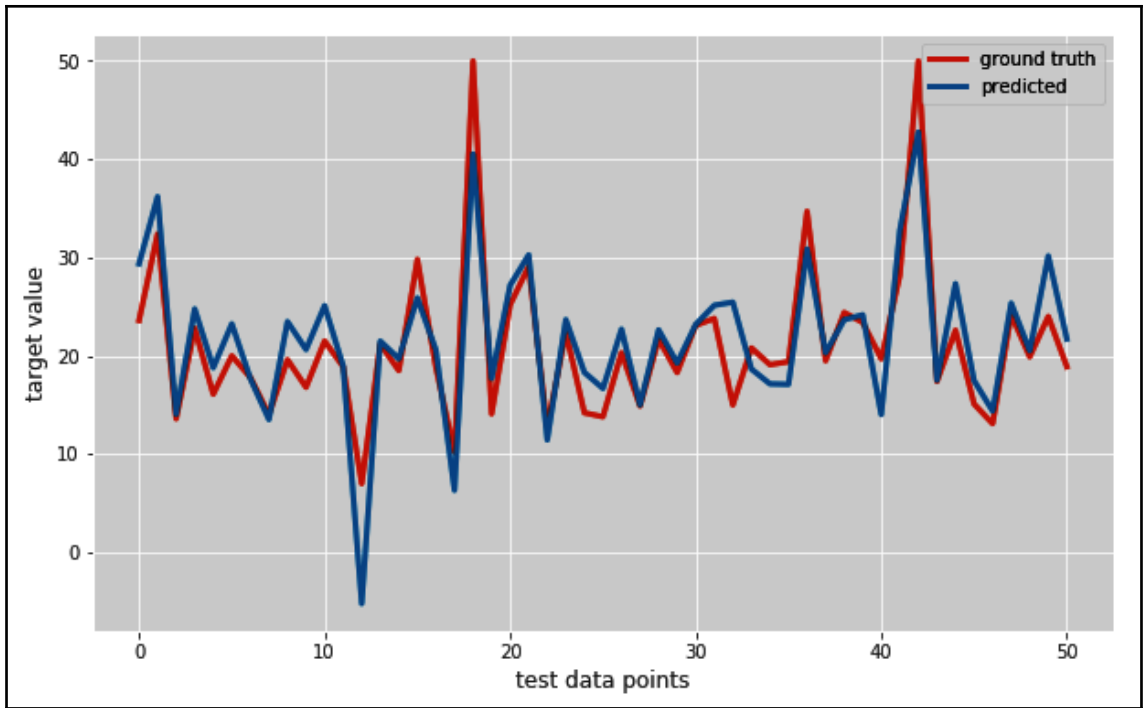


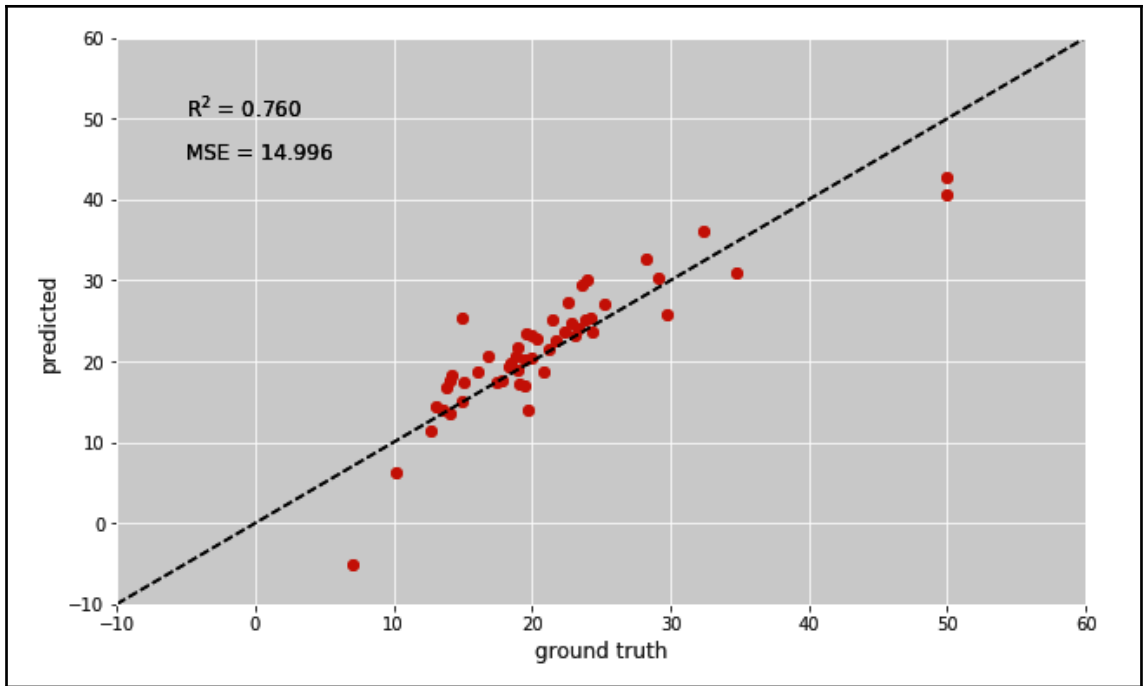


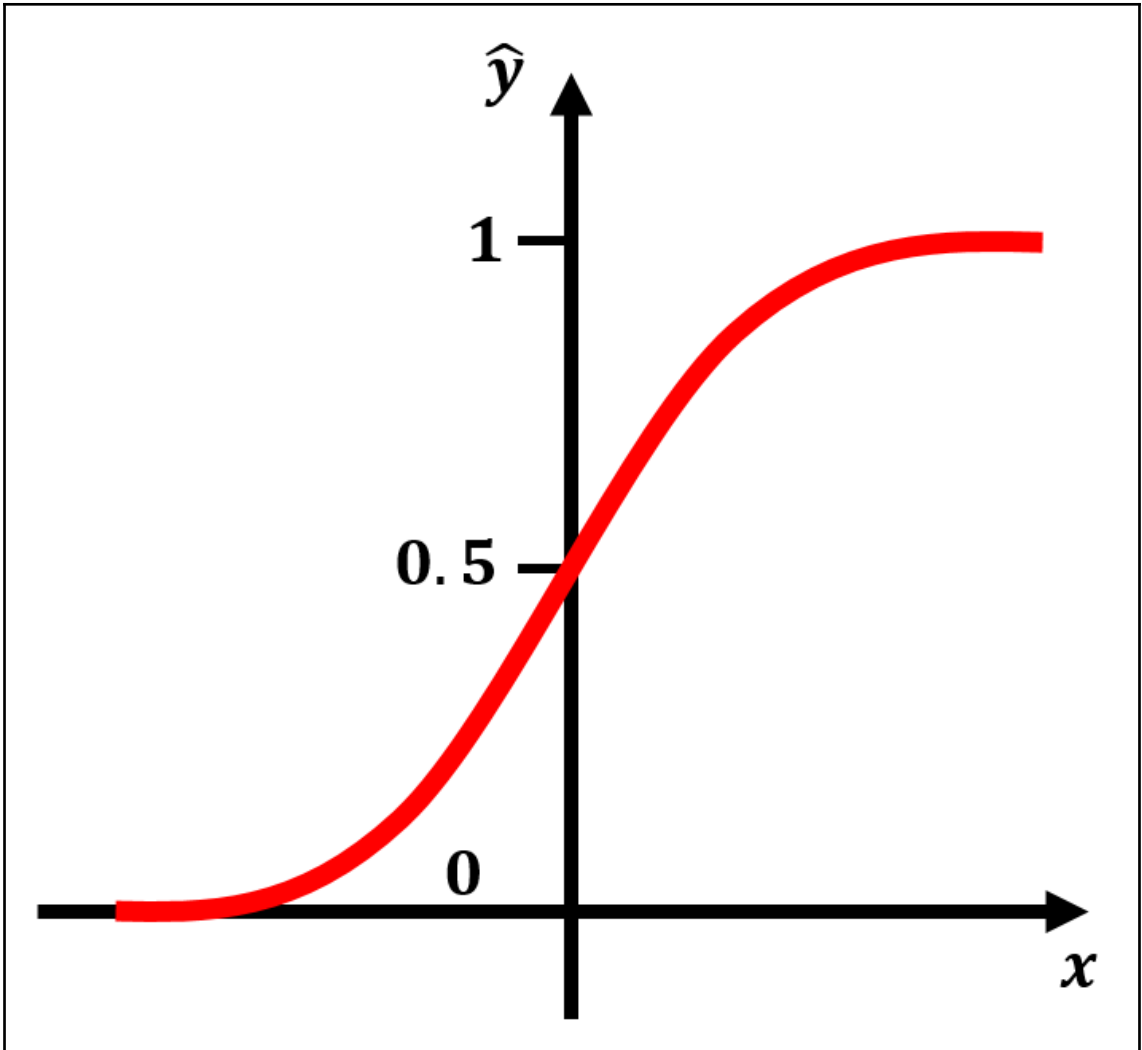


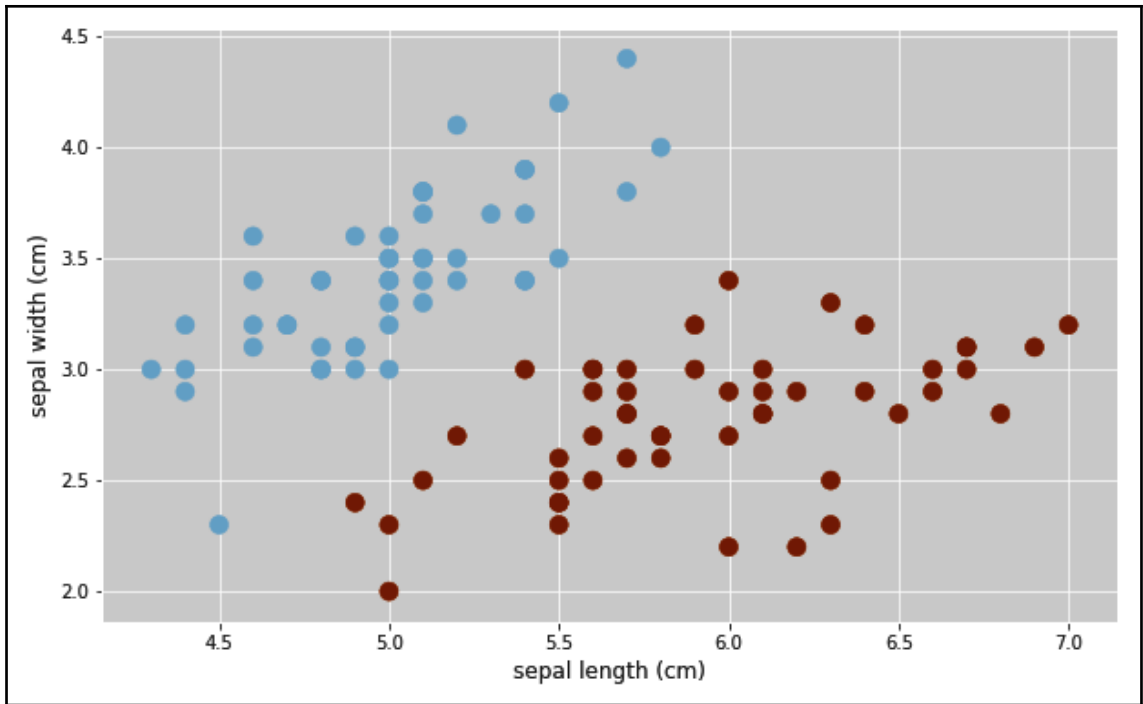




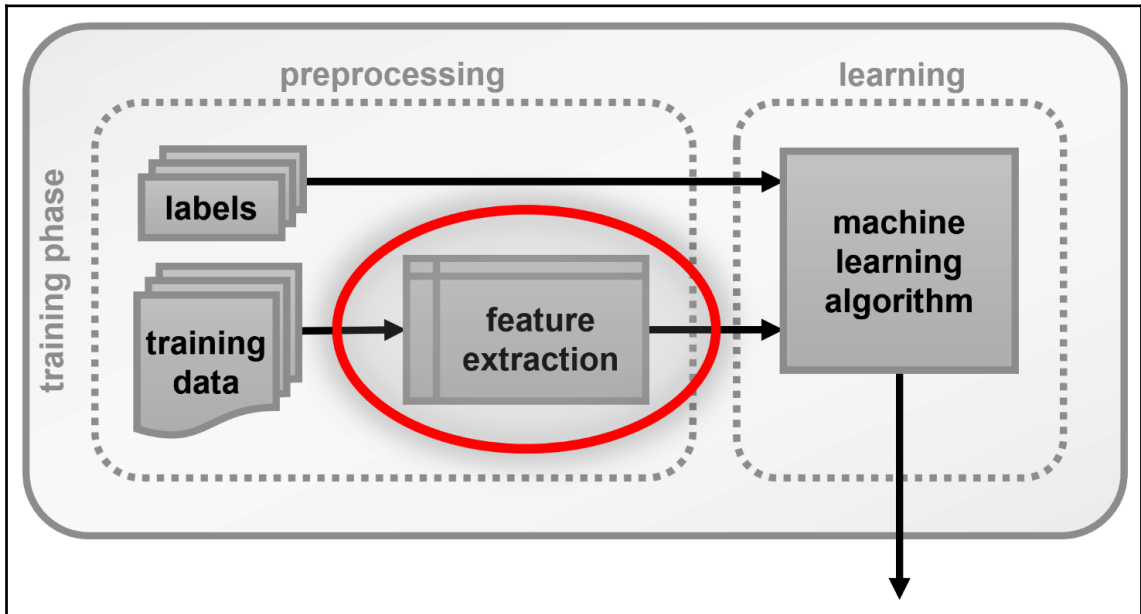


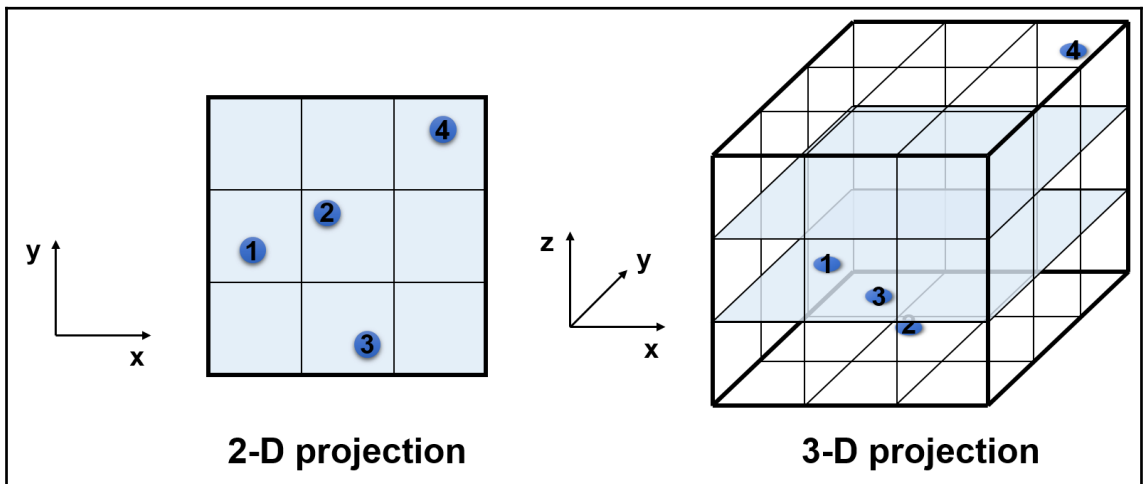
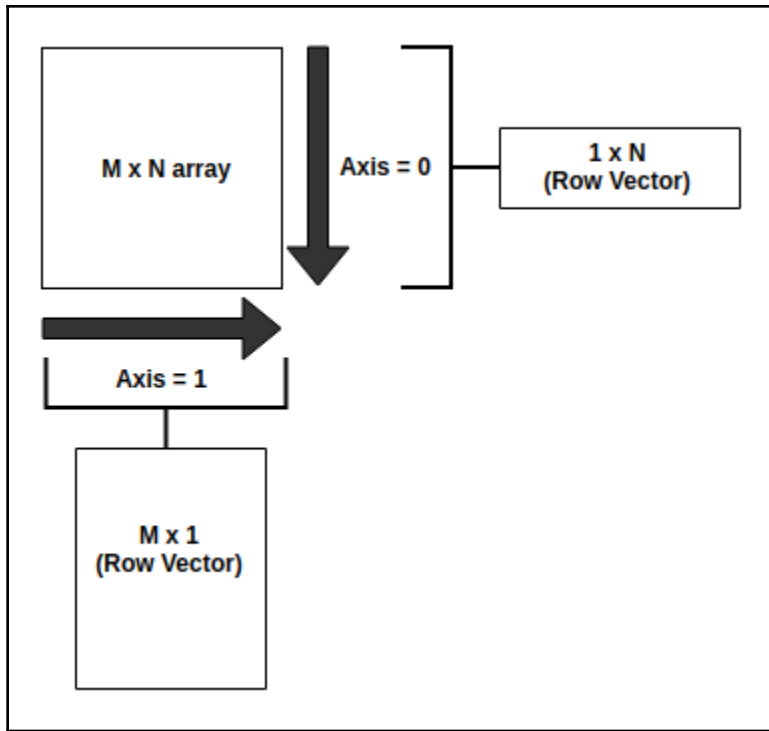


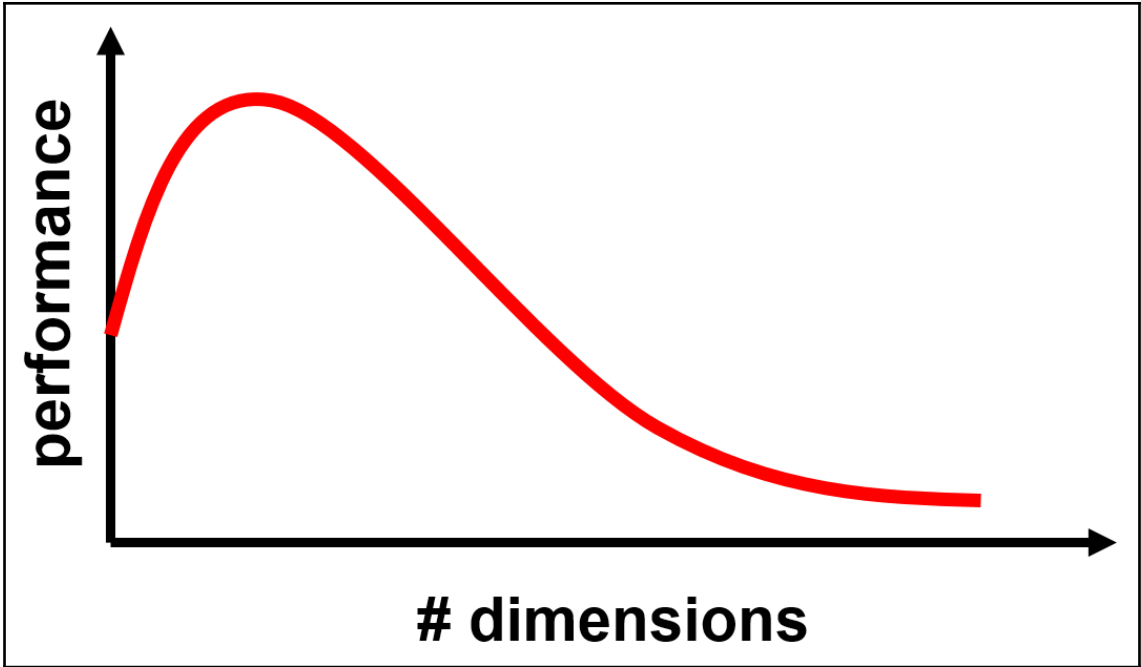


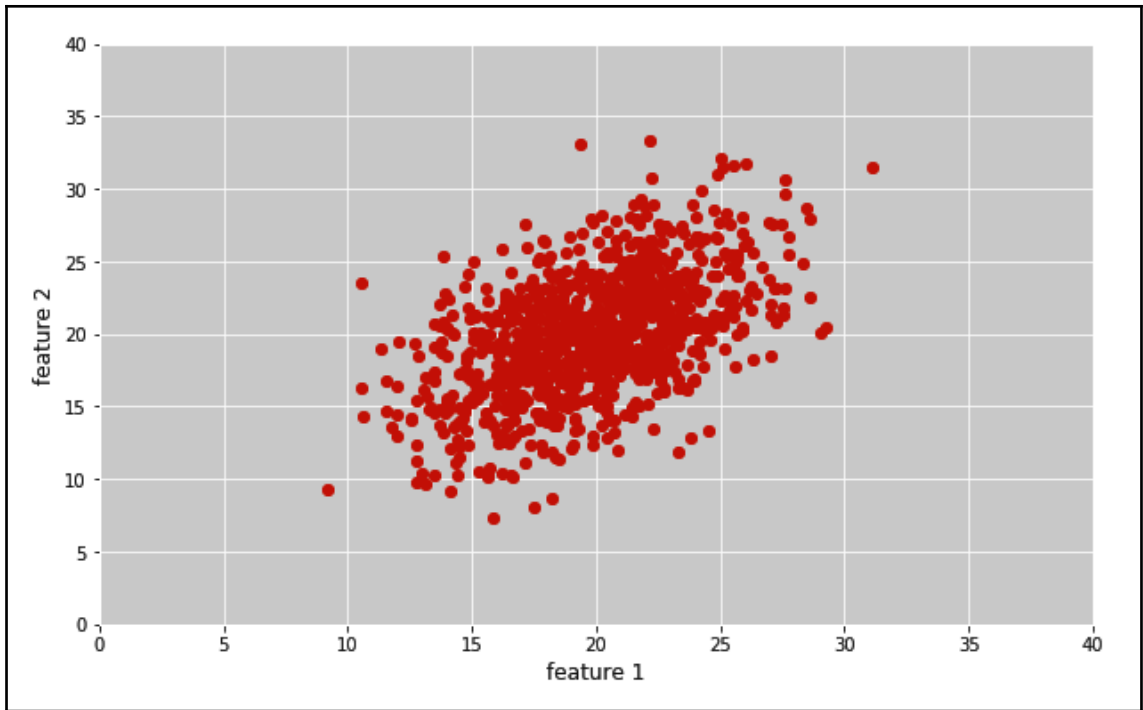


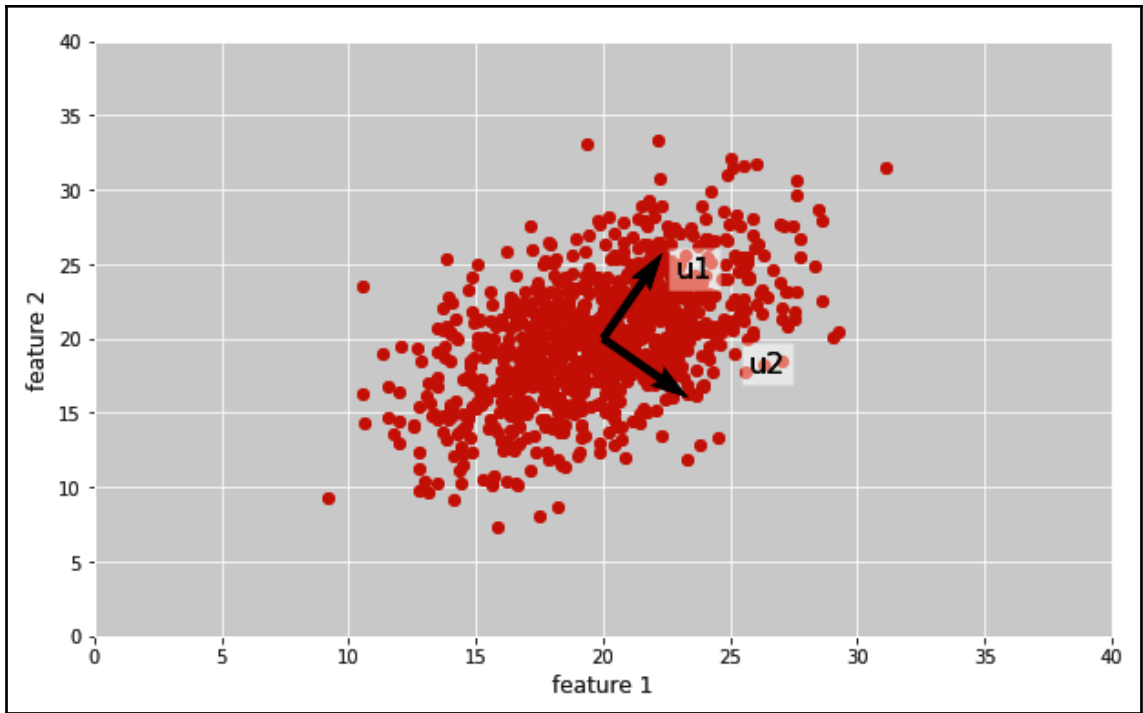
Chapter 4: Representing Data and Engineering Features

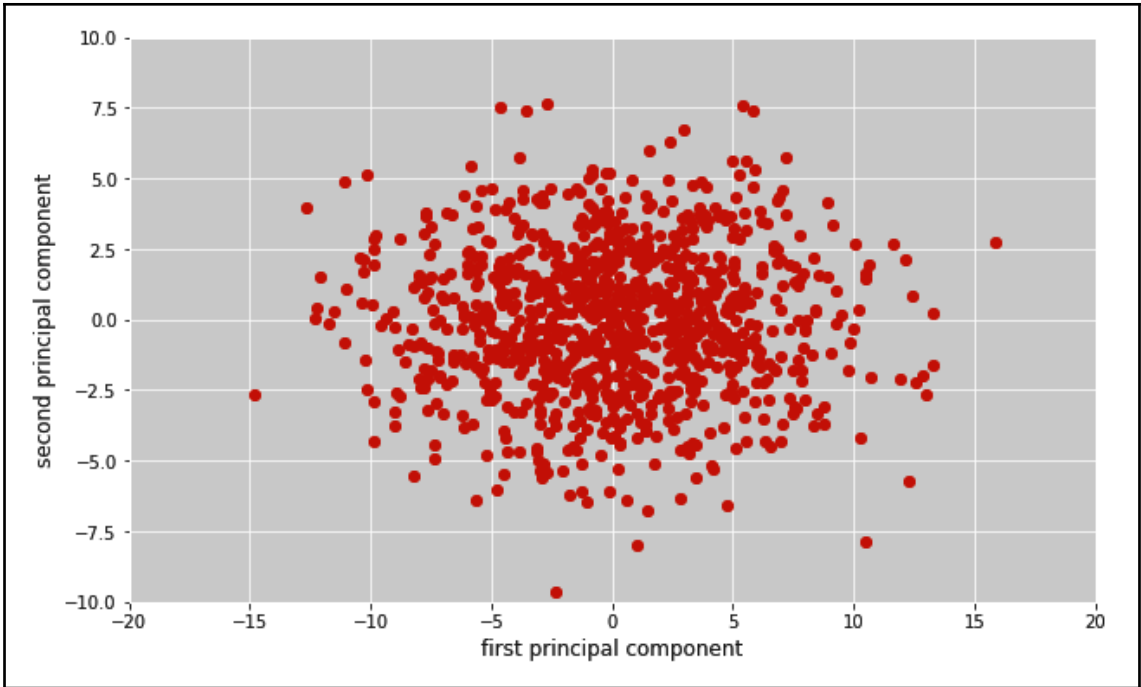


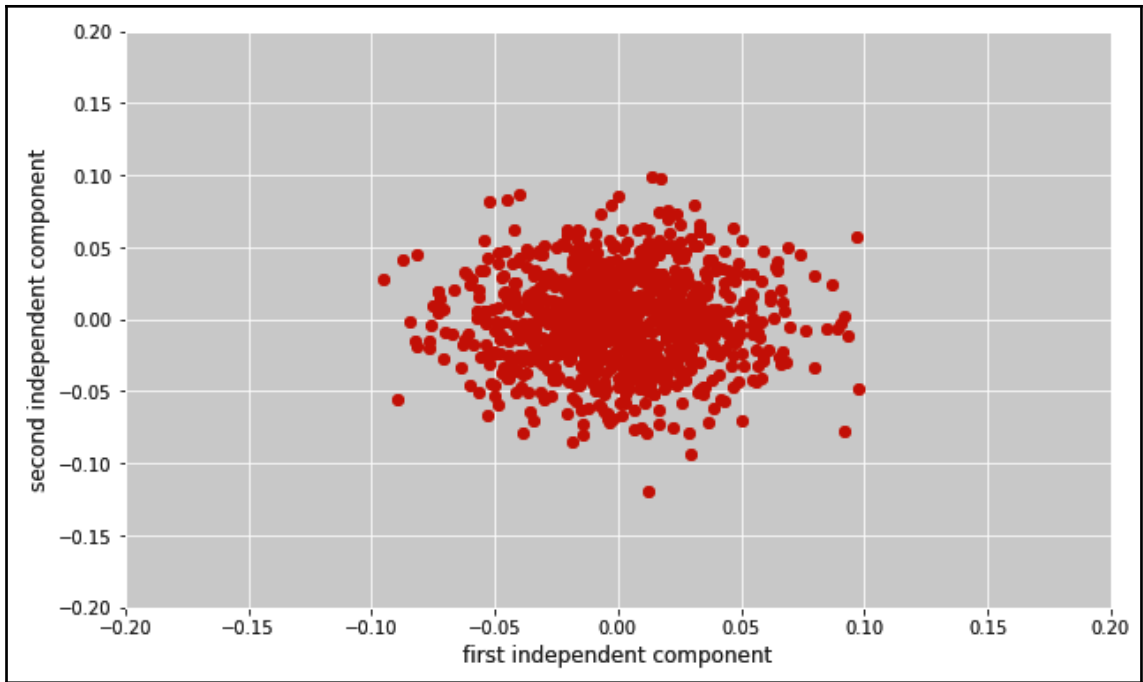


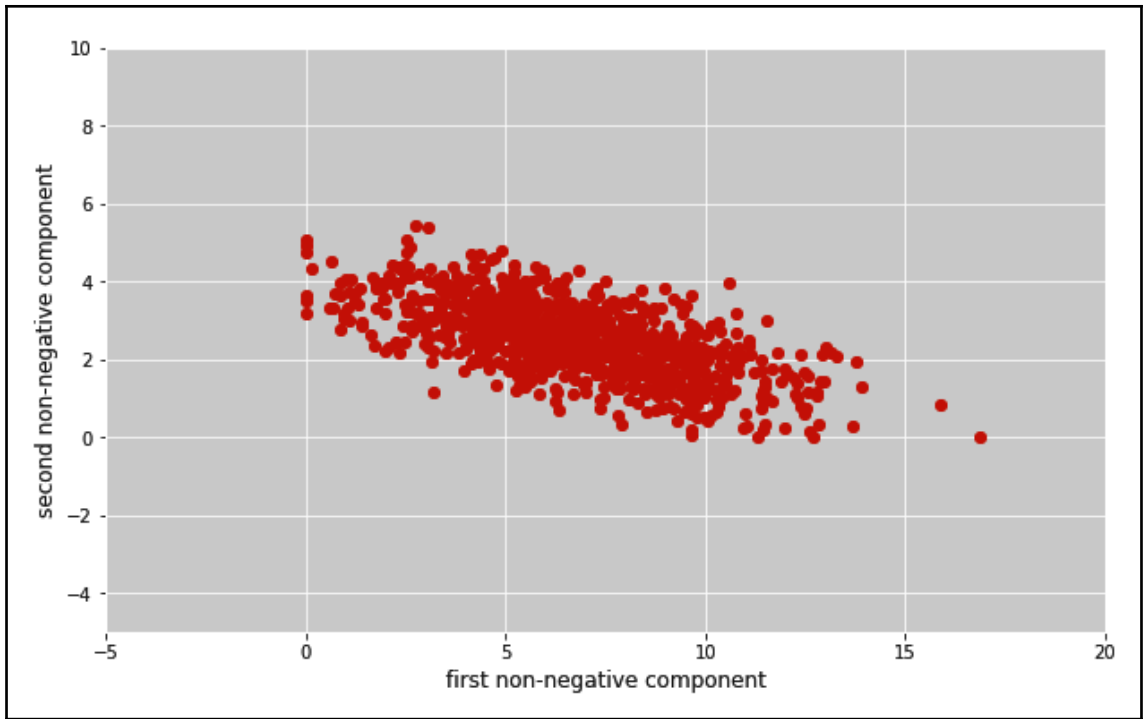


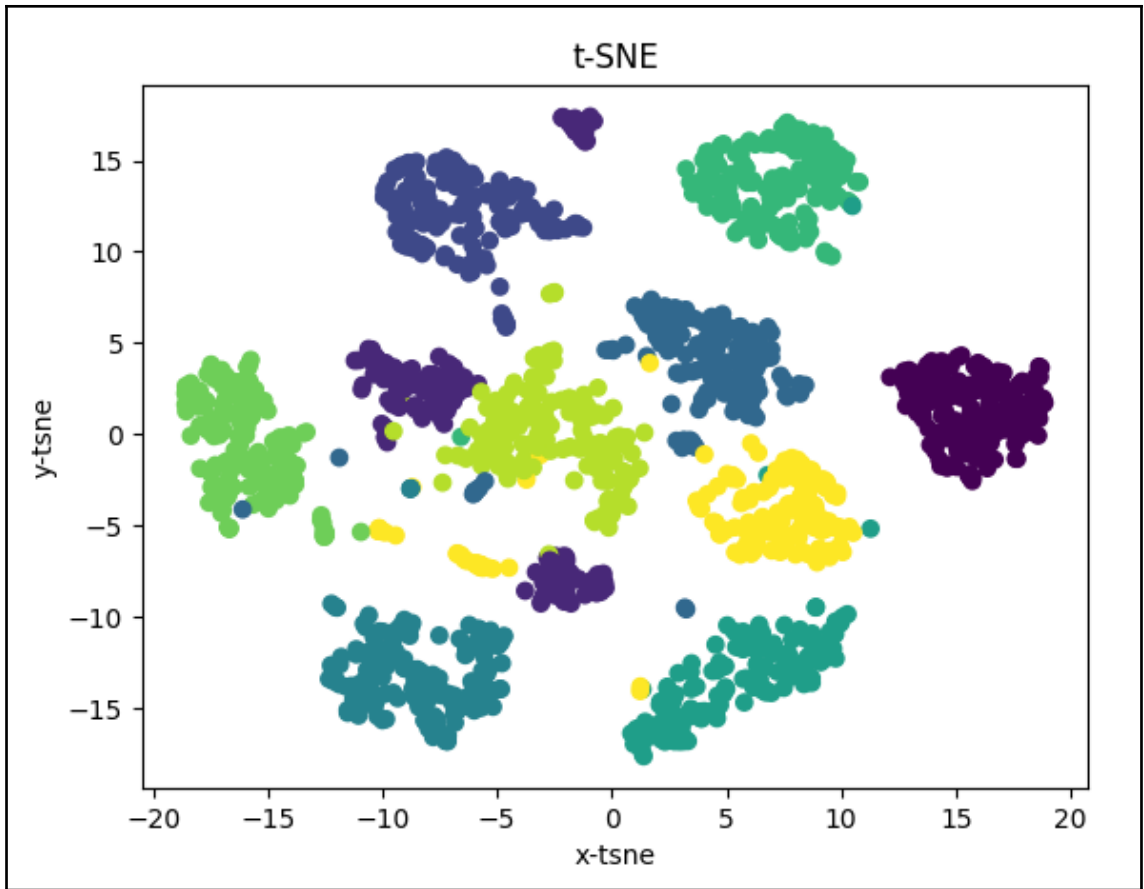


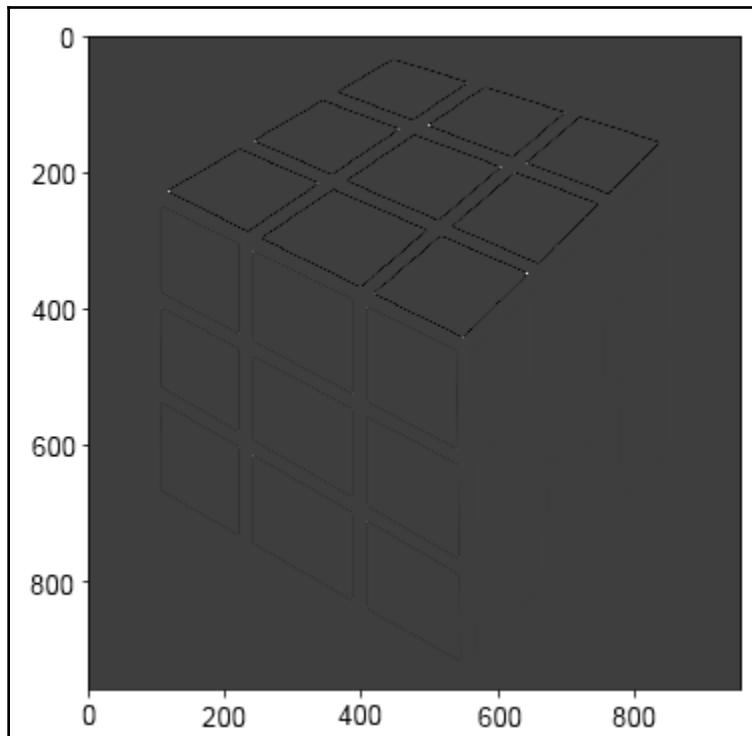
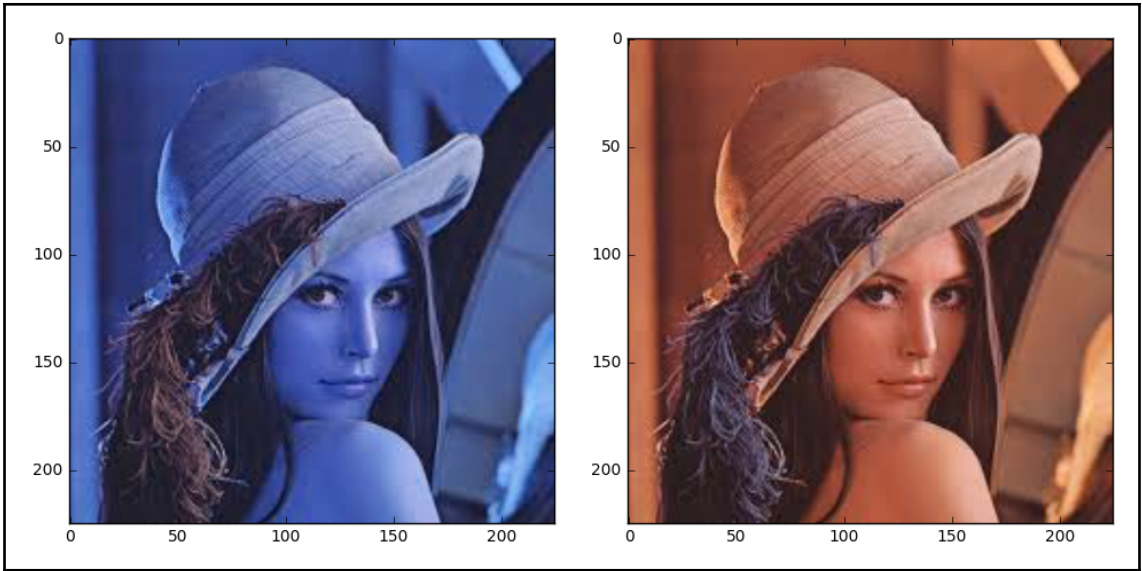


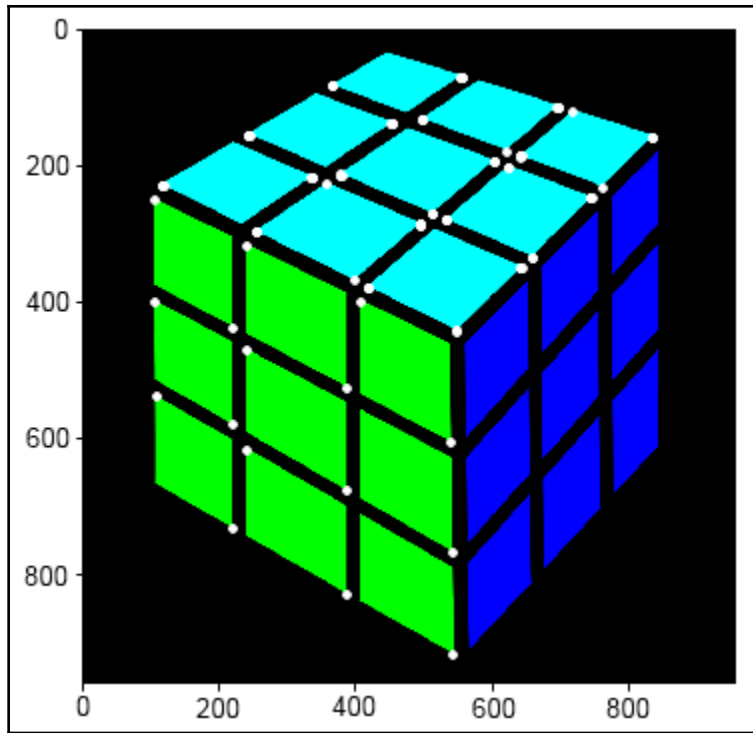


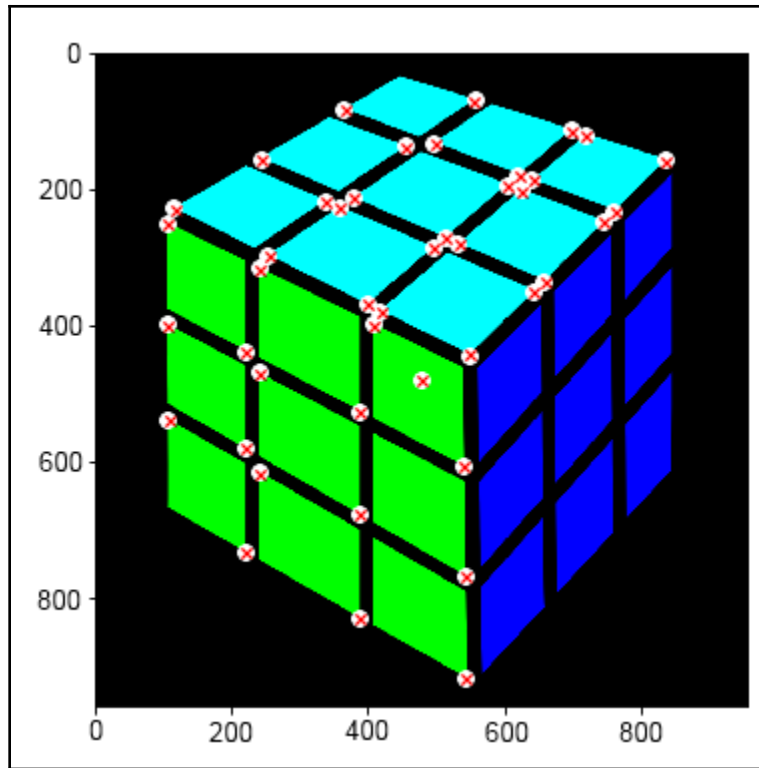


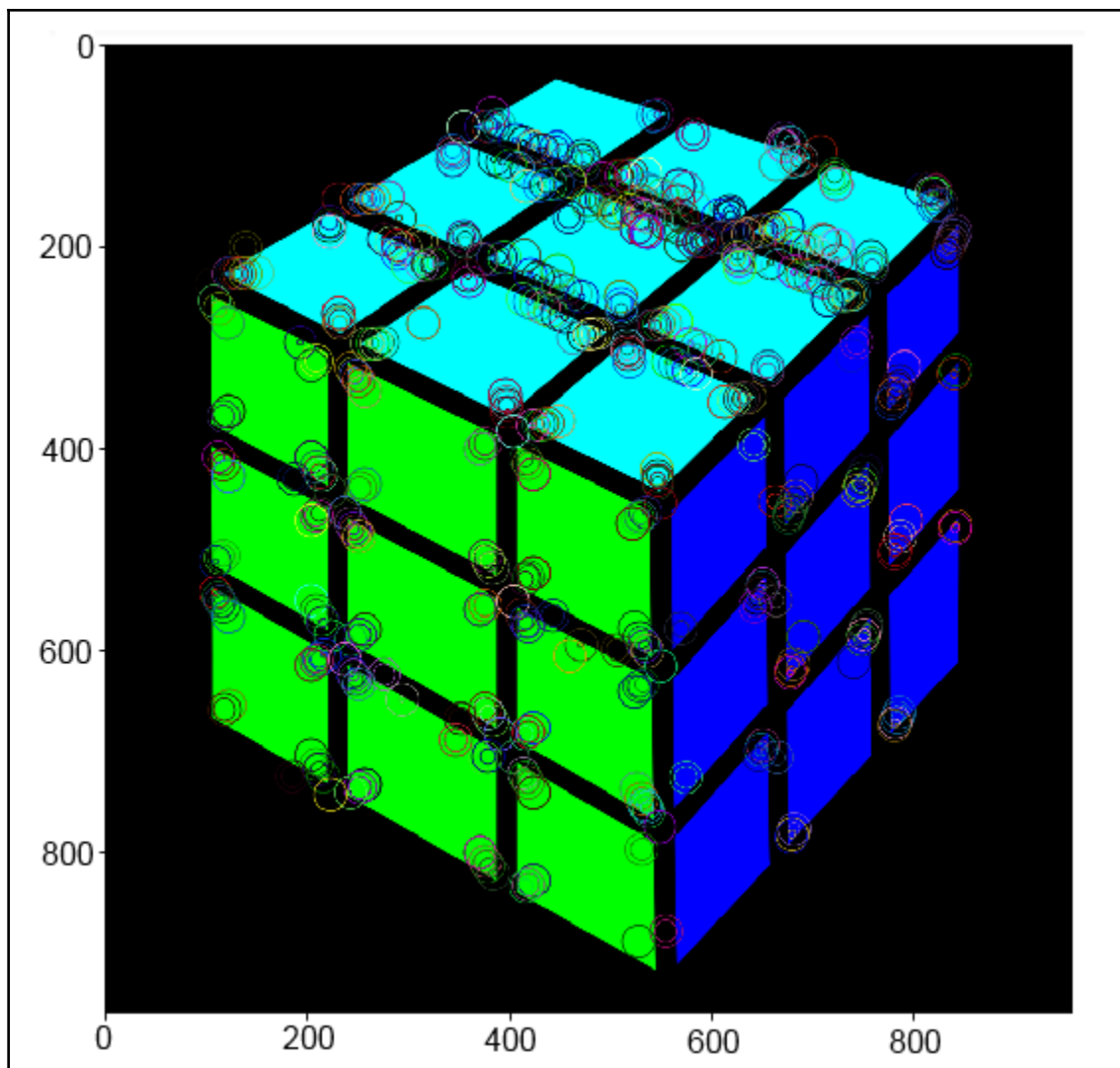


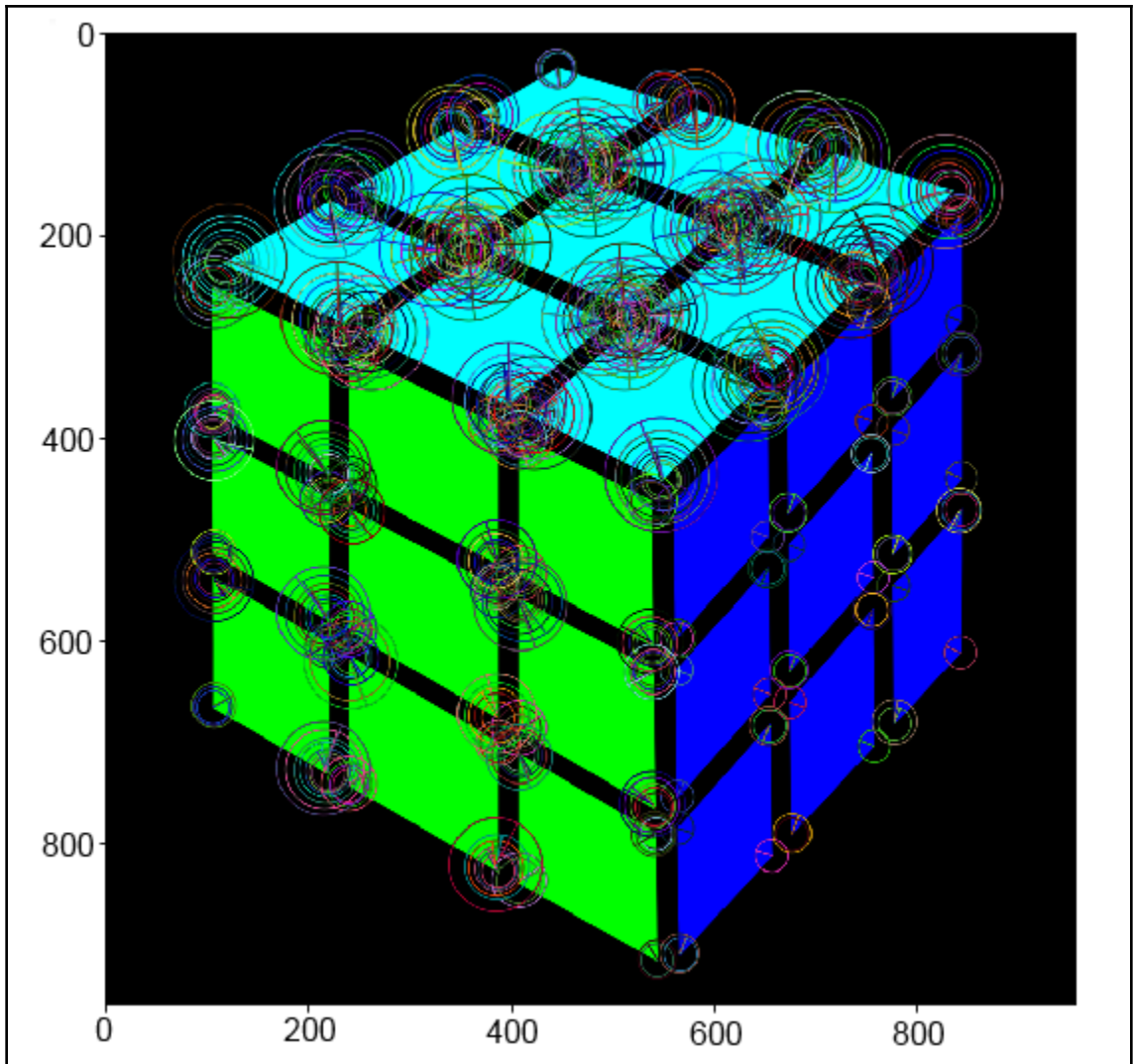




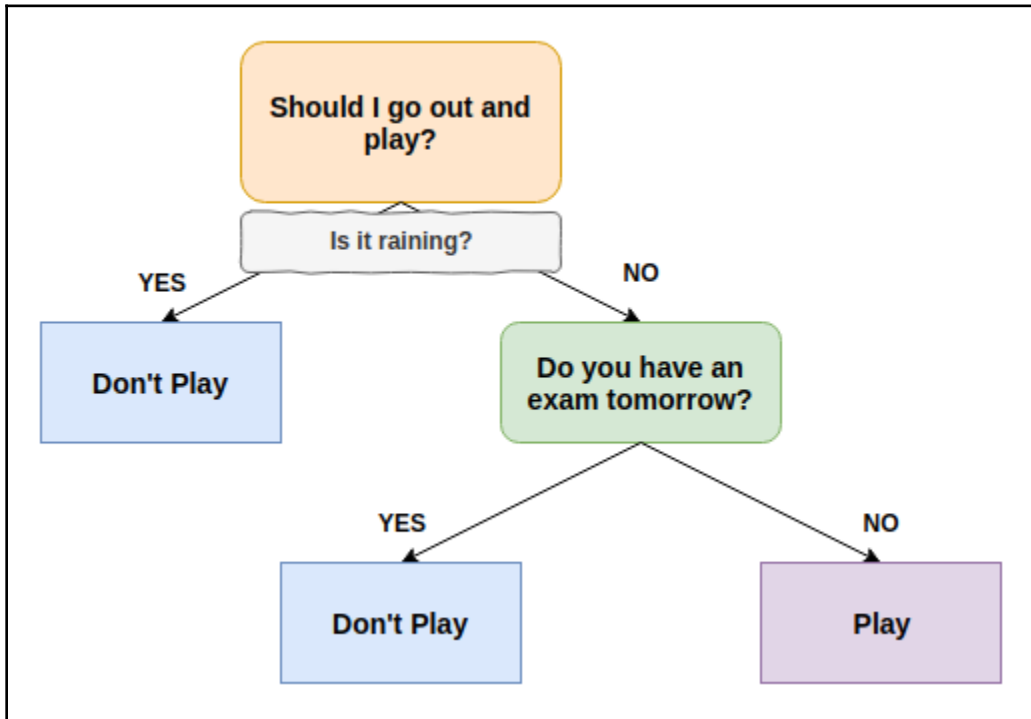


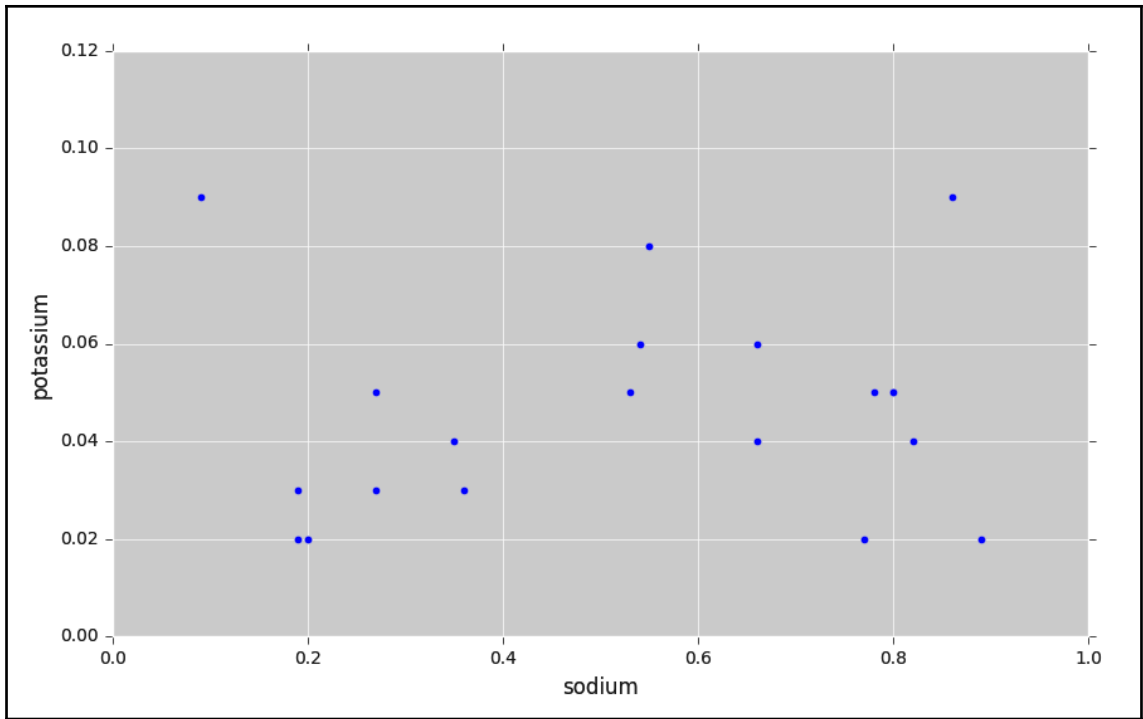


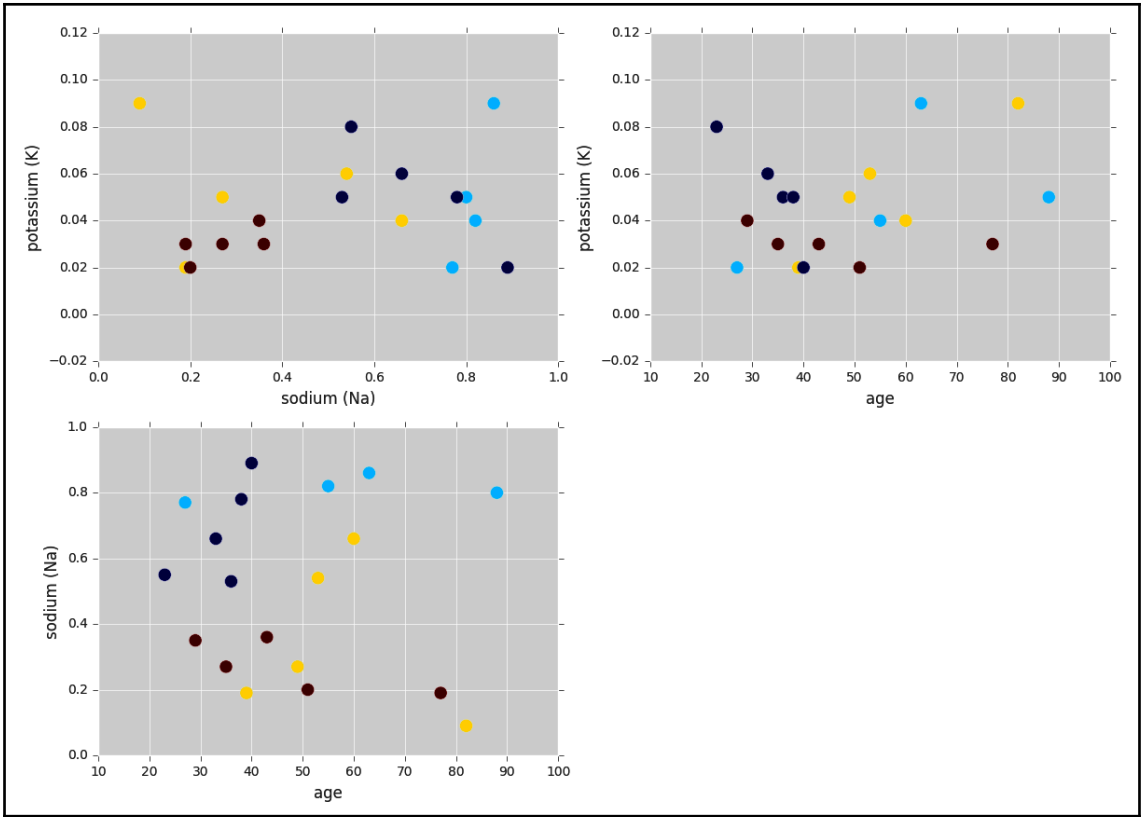


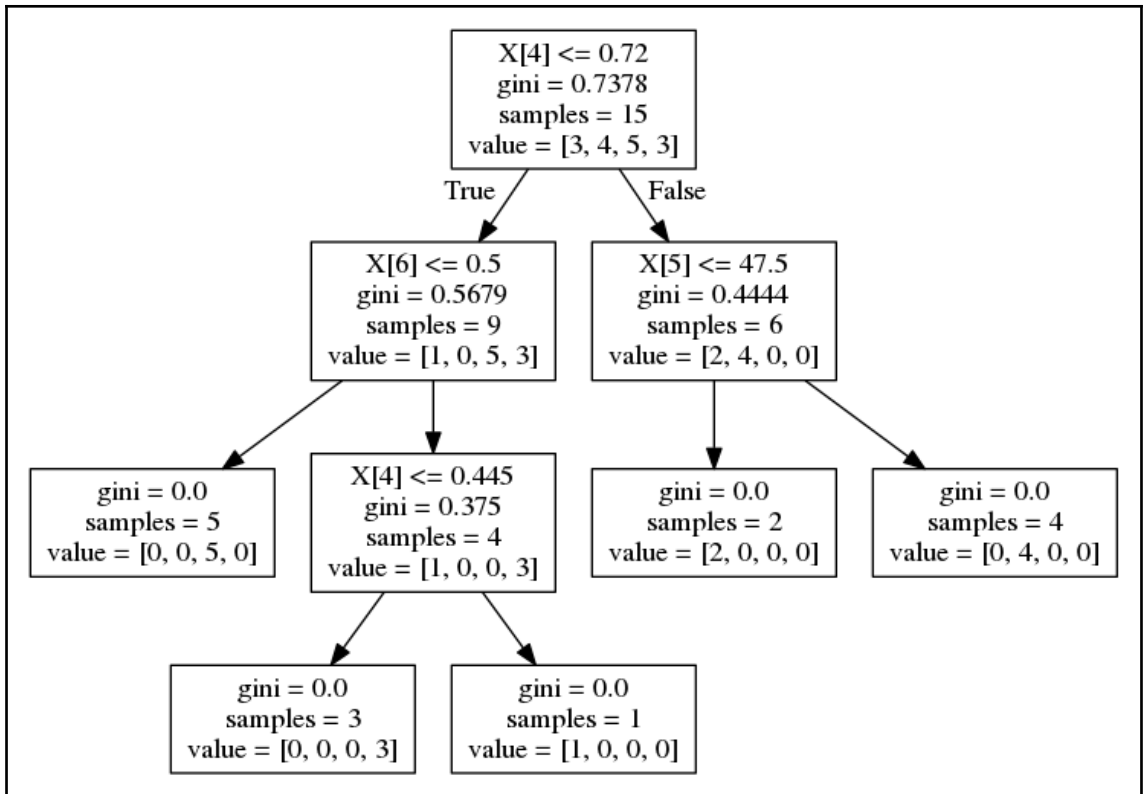


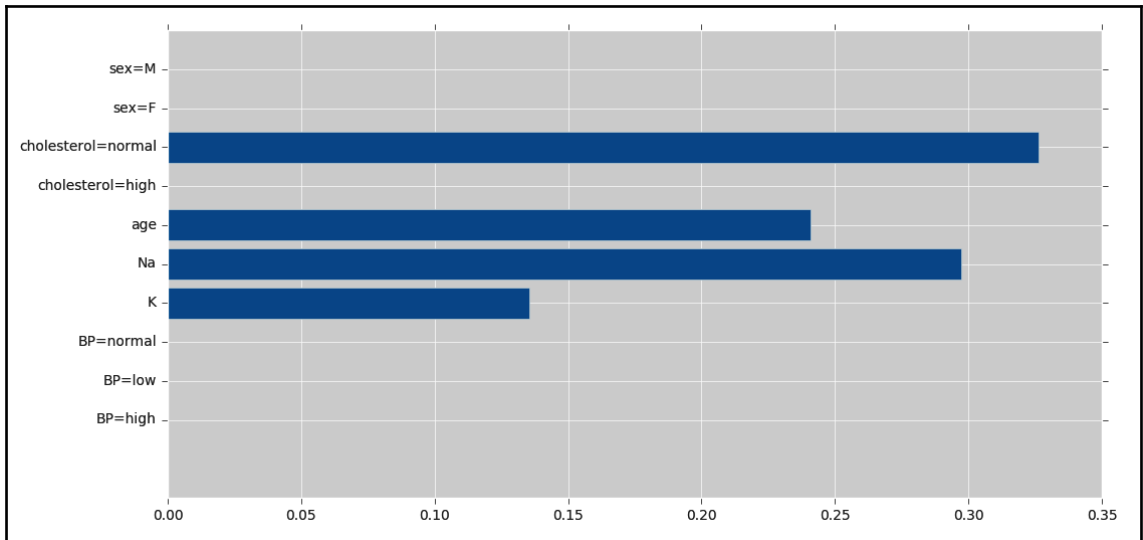
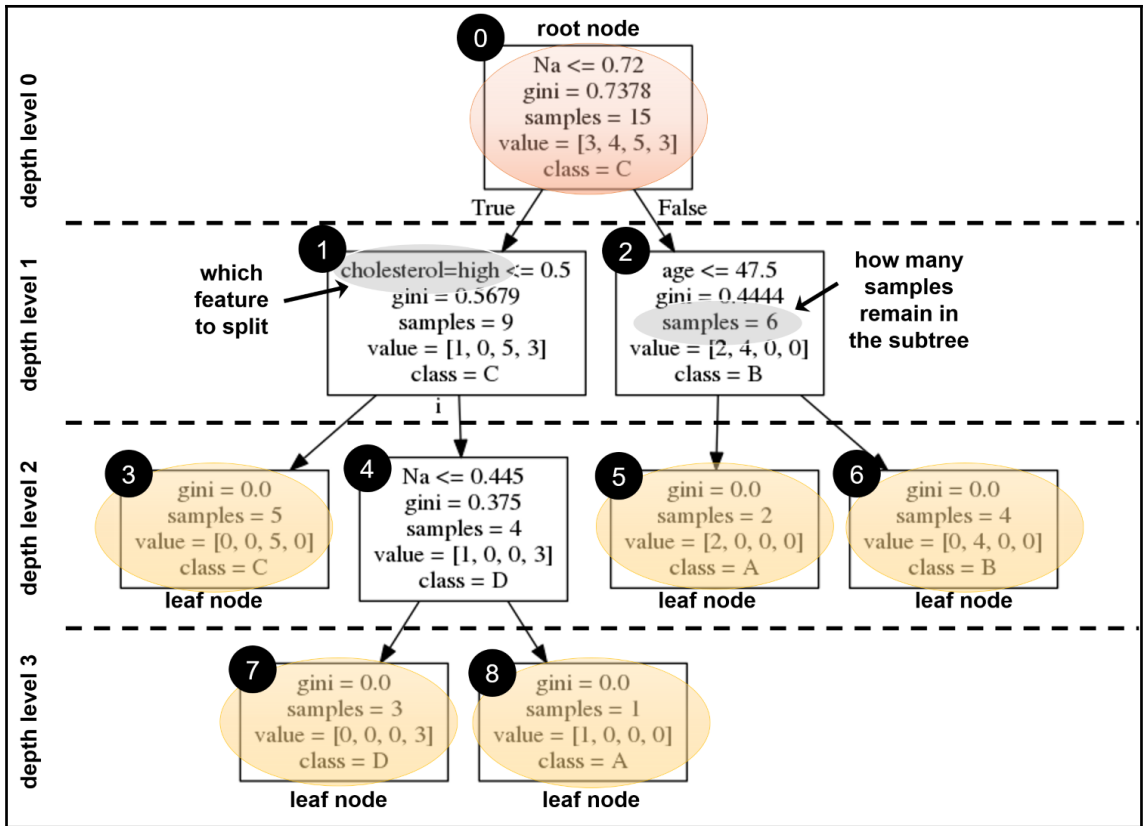
Chapter 5: Using Decision Trees to Make a Medical Diagnosis

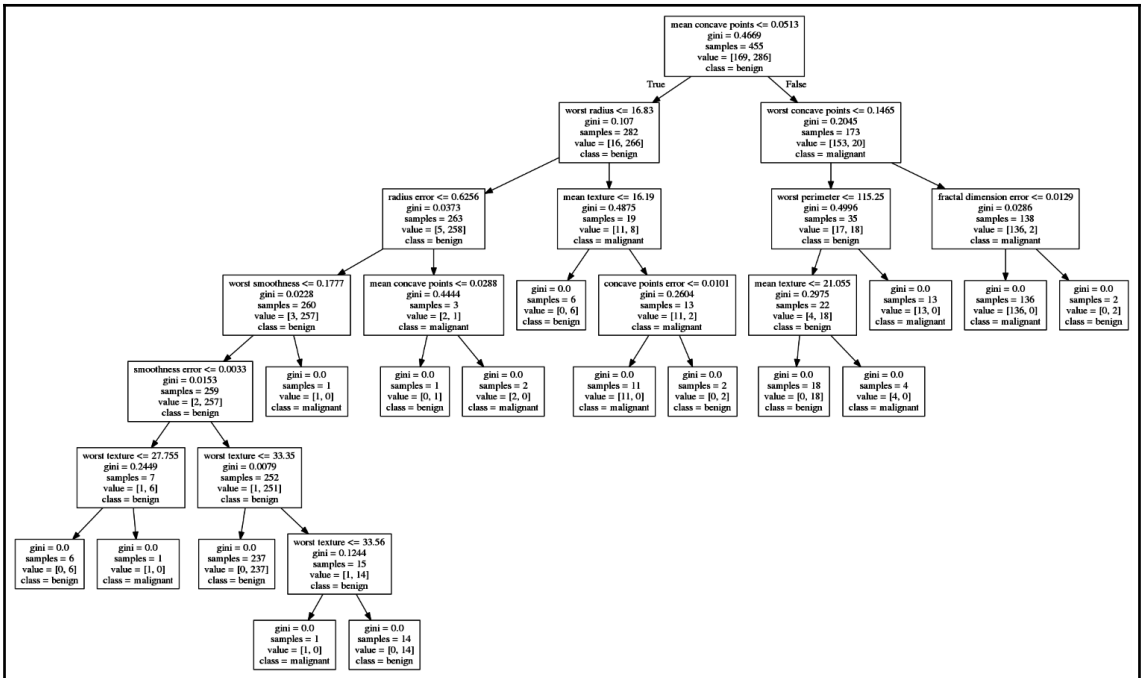
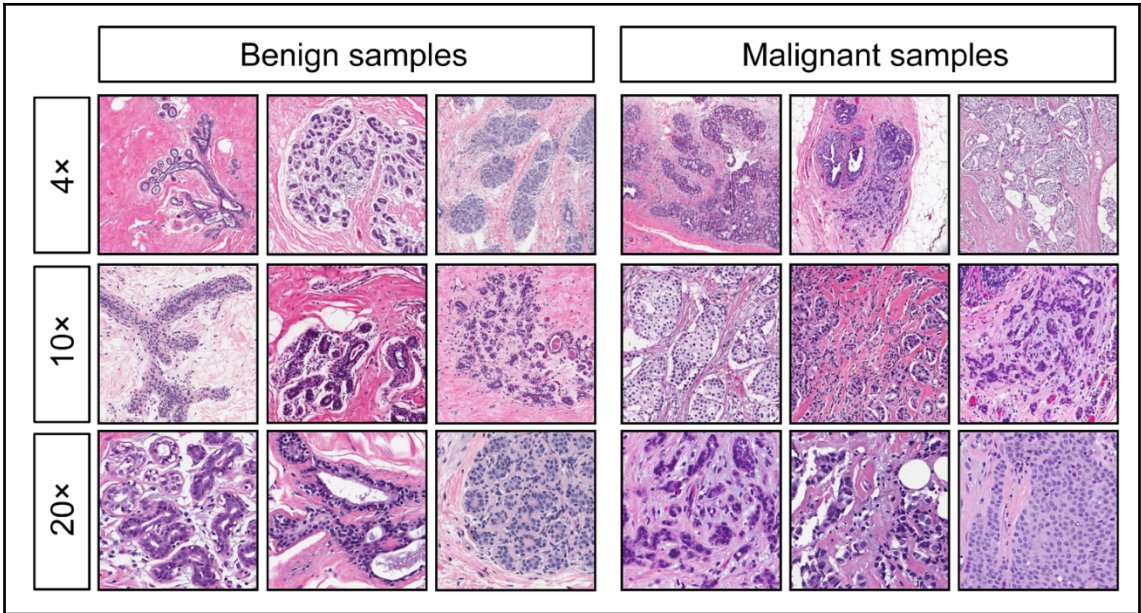


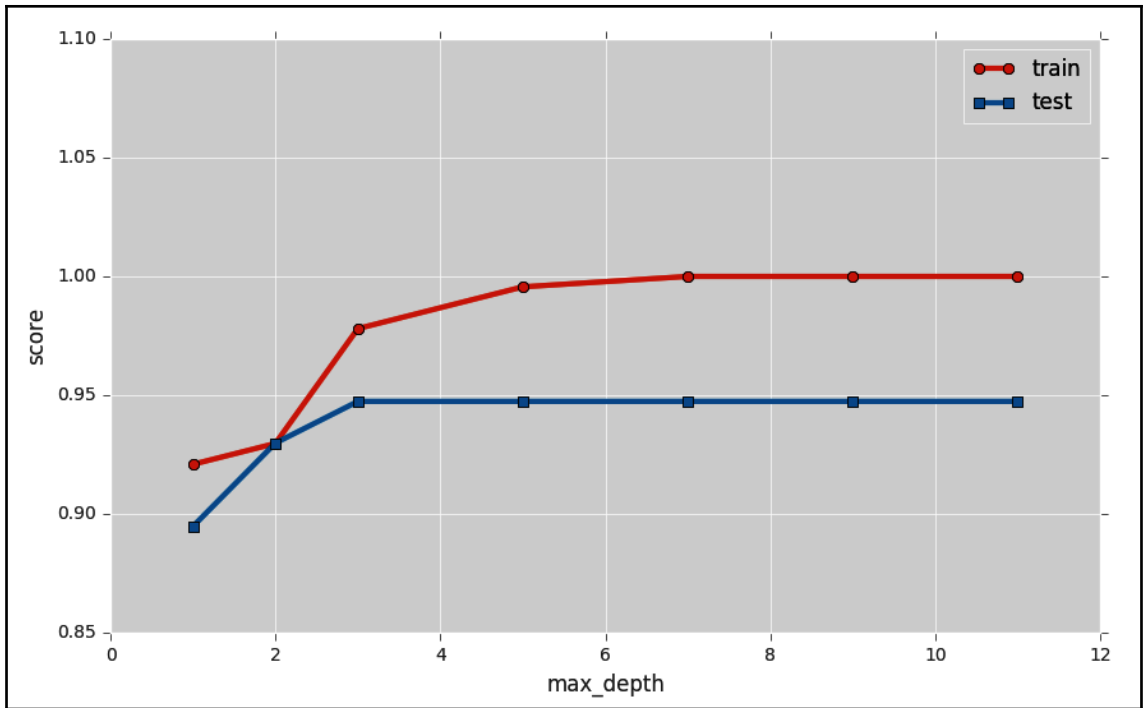


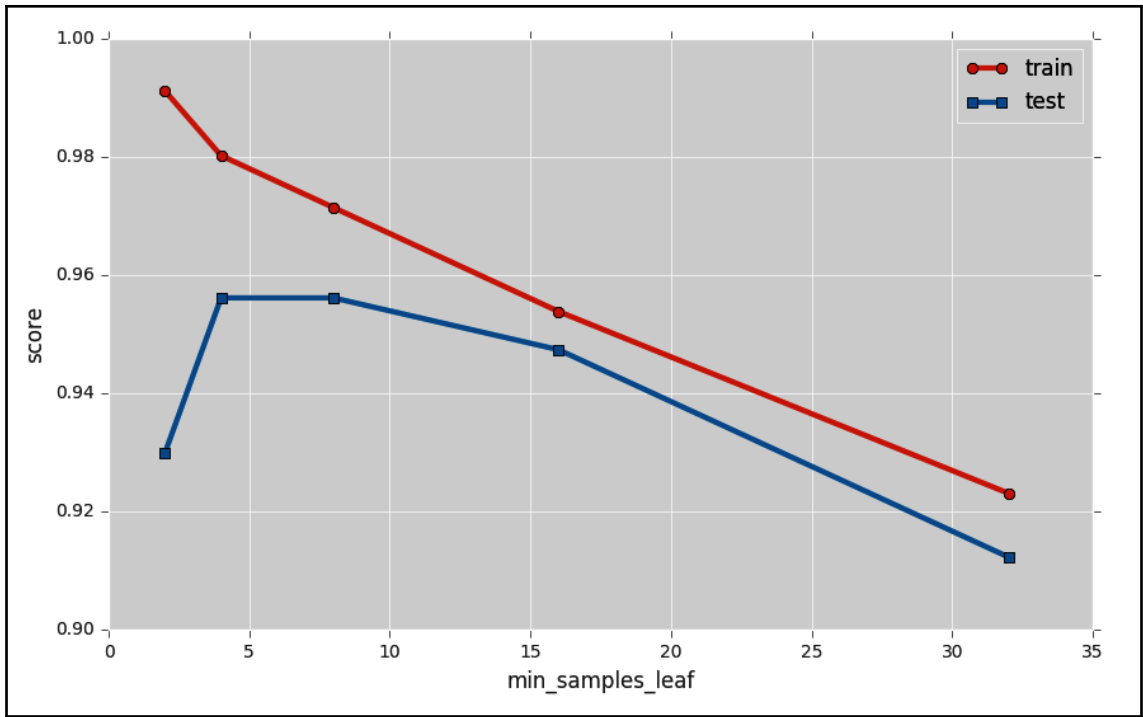


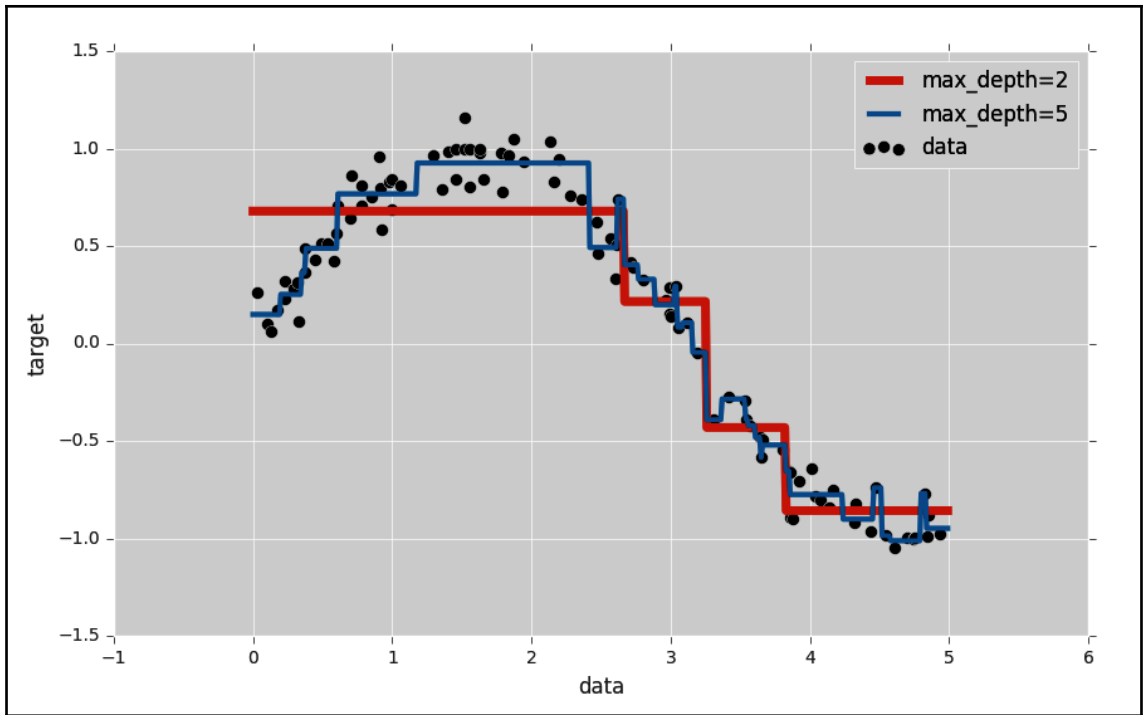




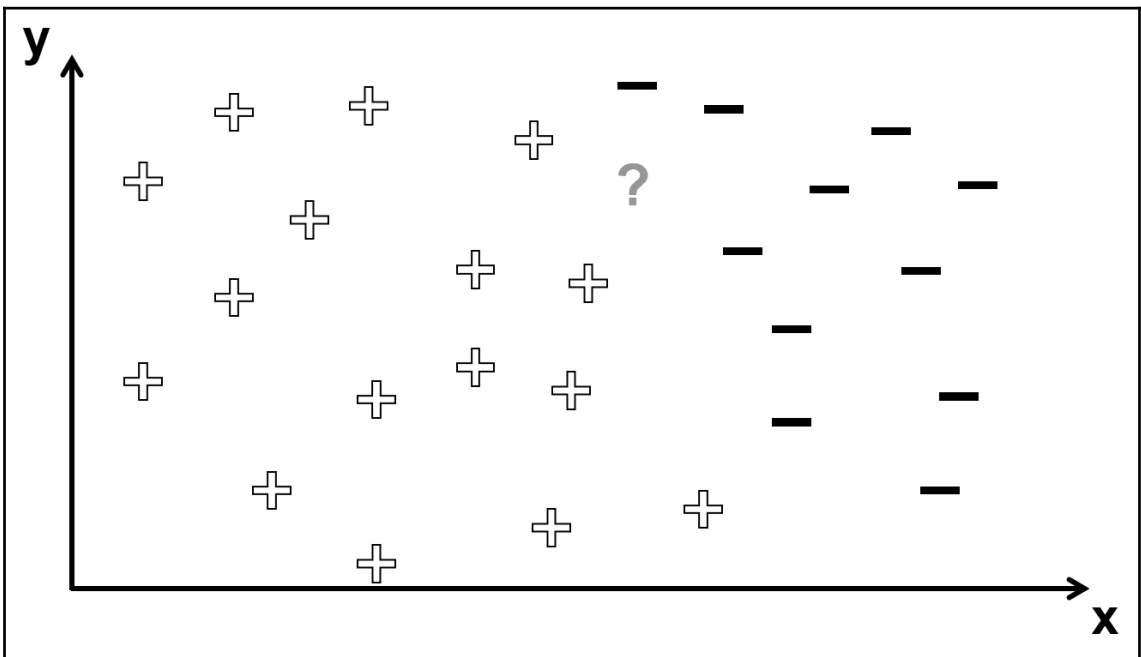
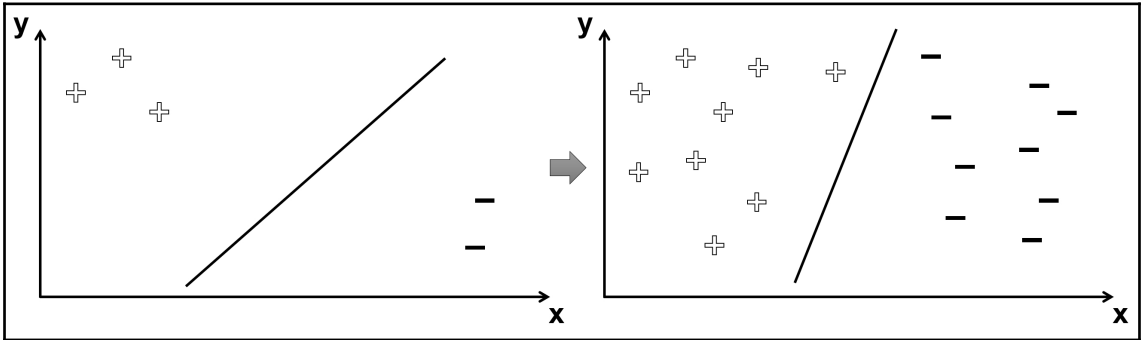


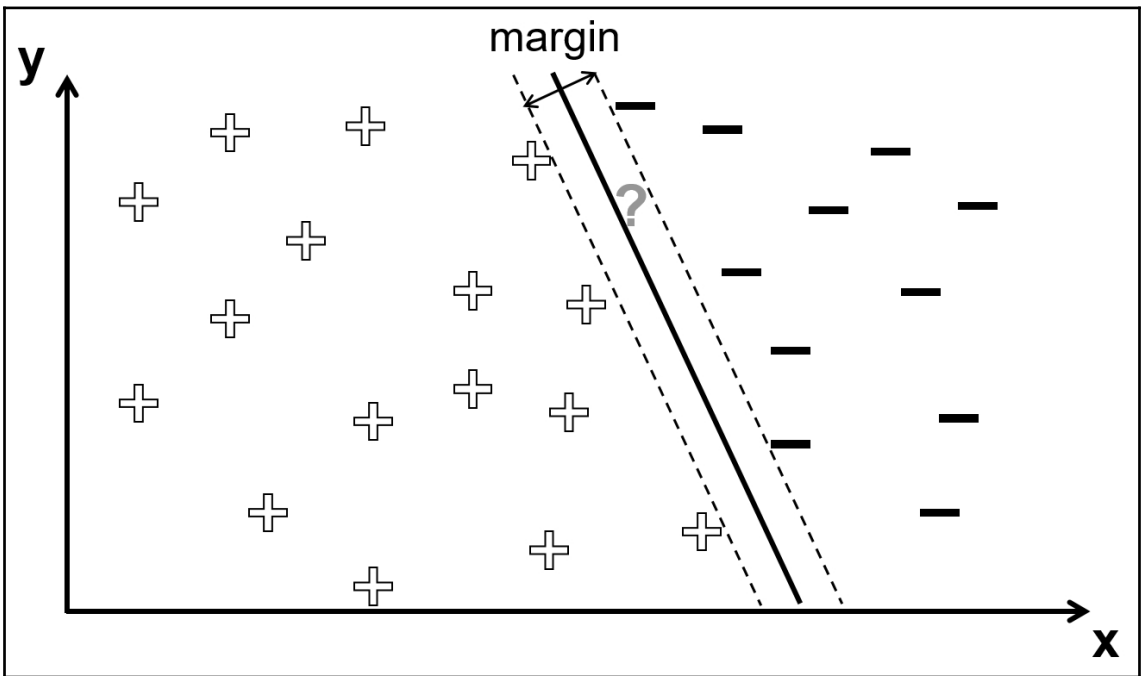
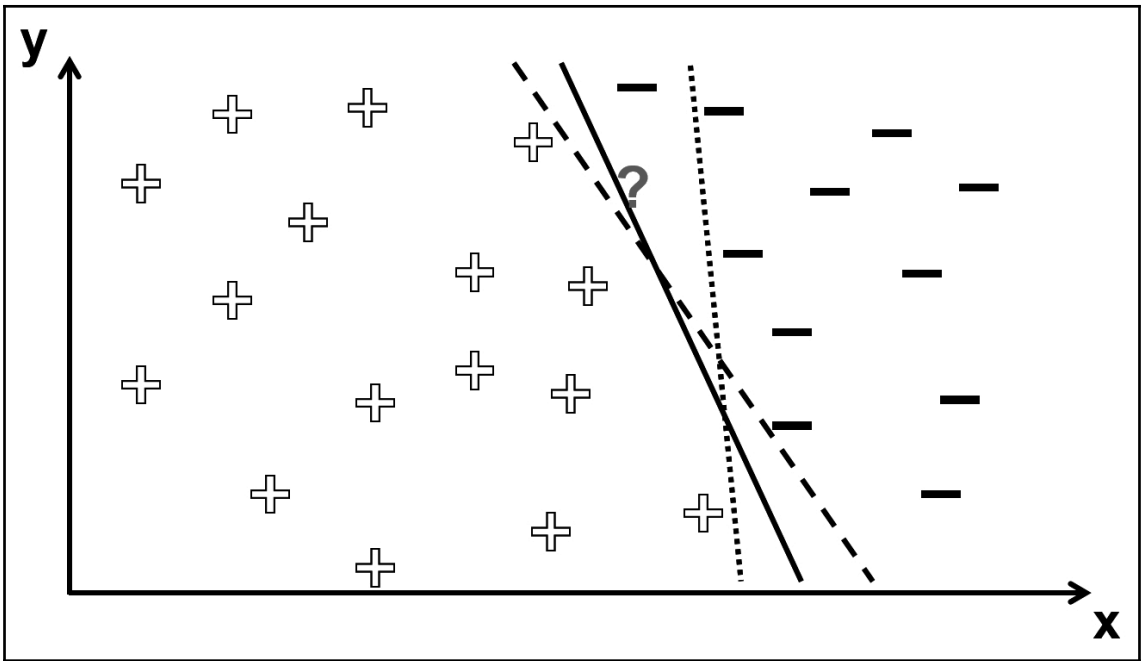


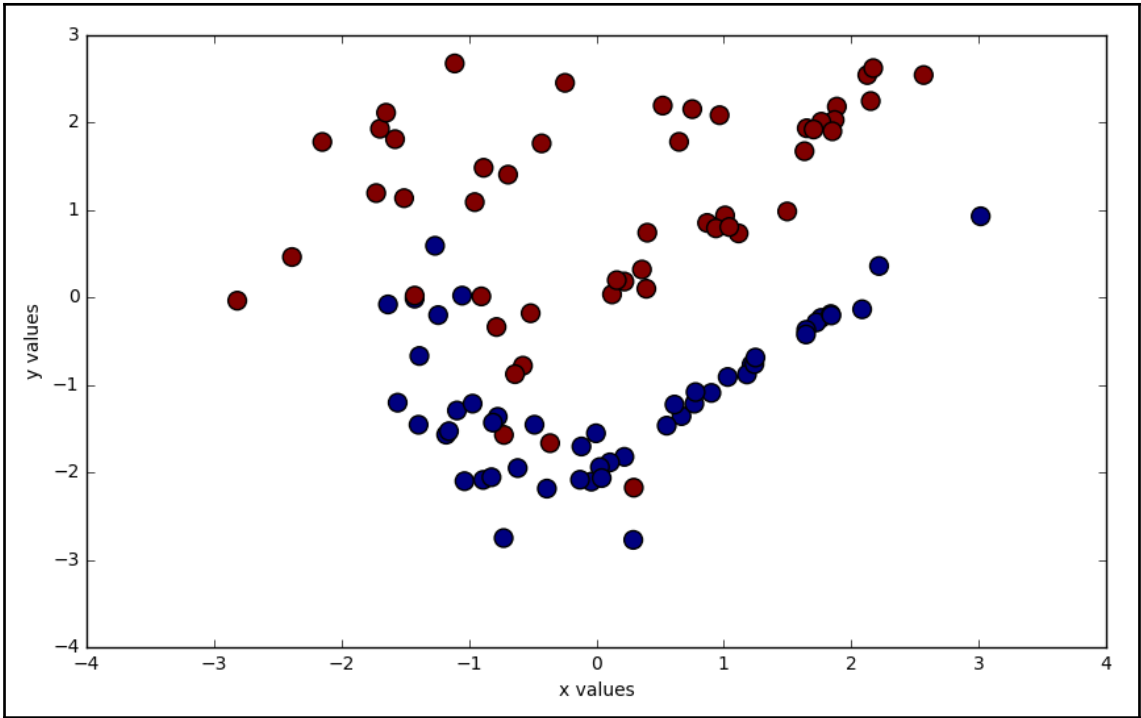


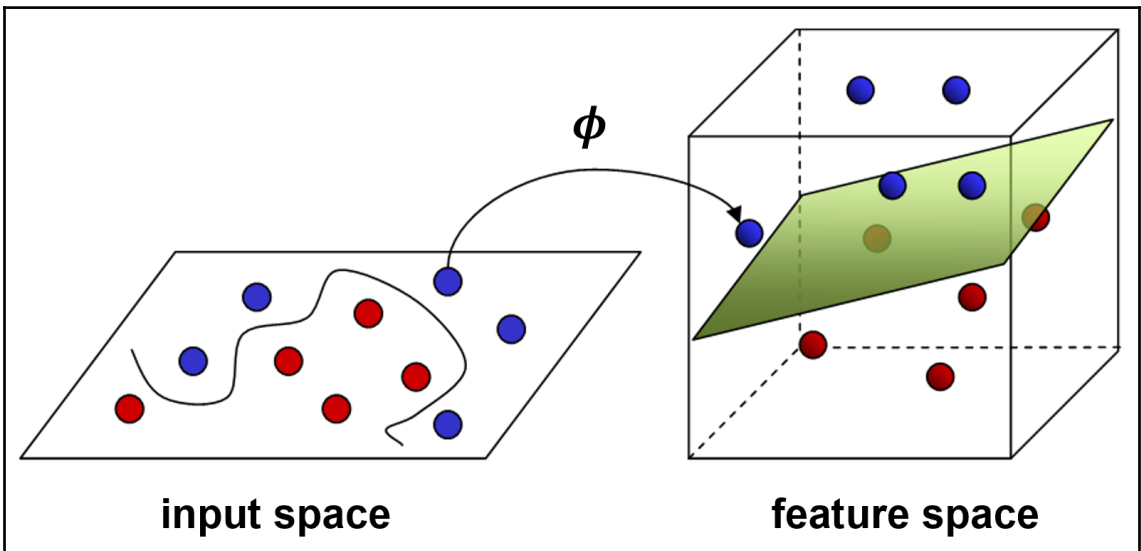
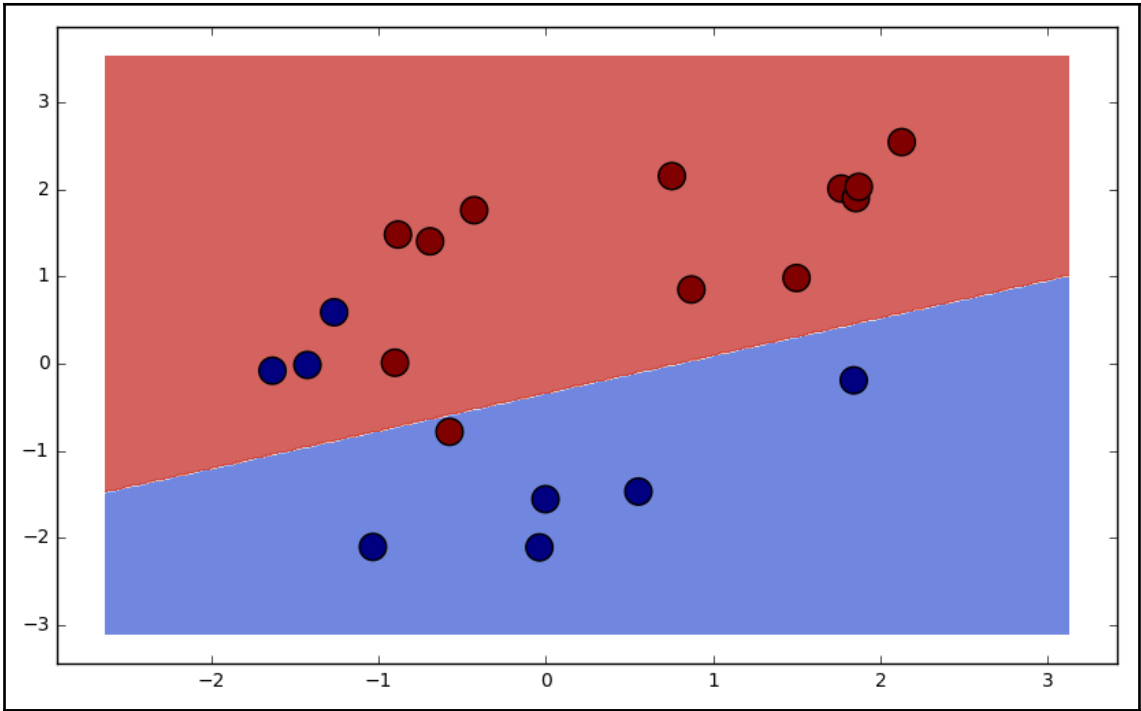


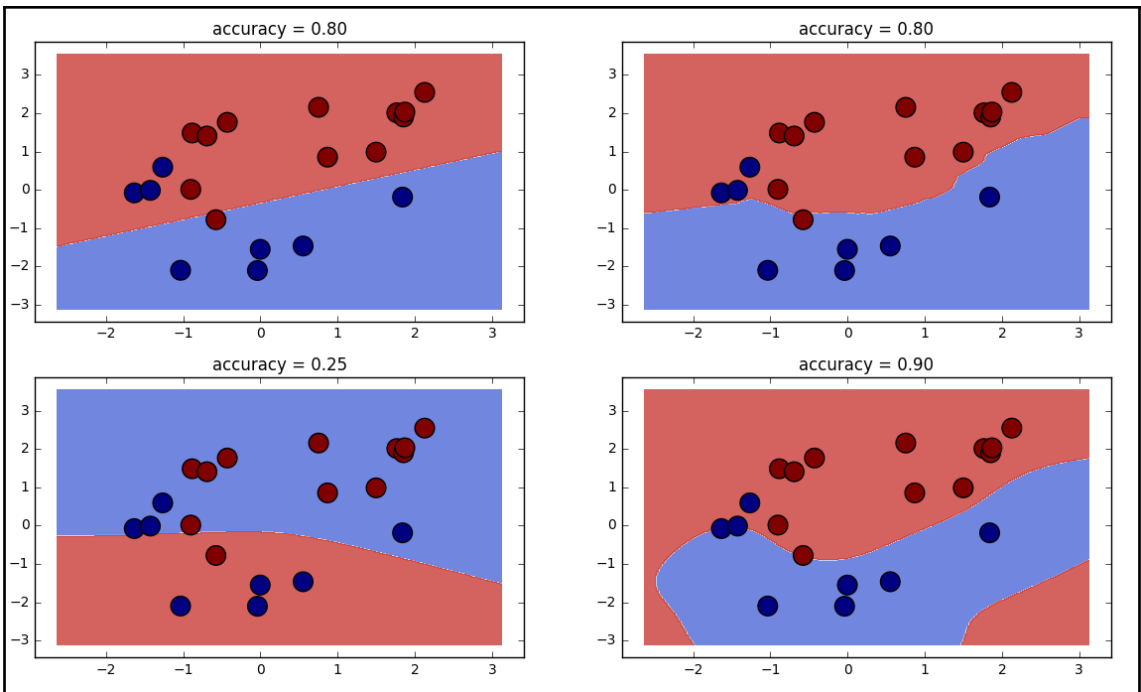
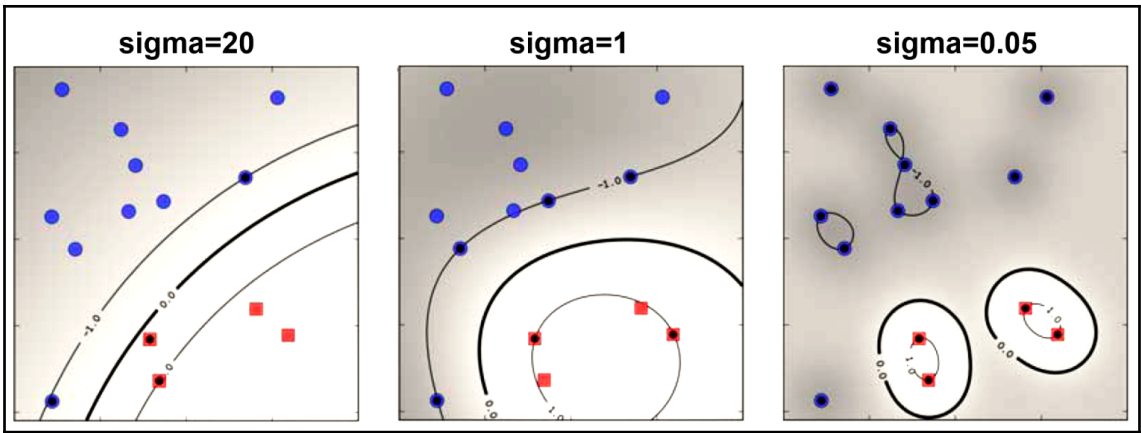
Chapter 6: Detecting Pedestrians with Support Vector Machines

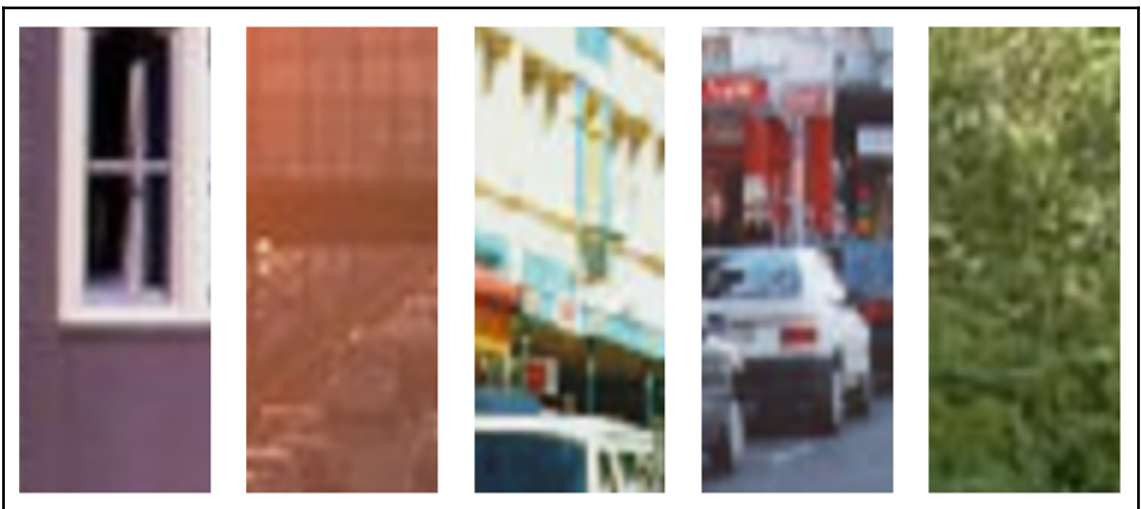


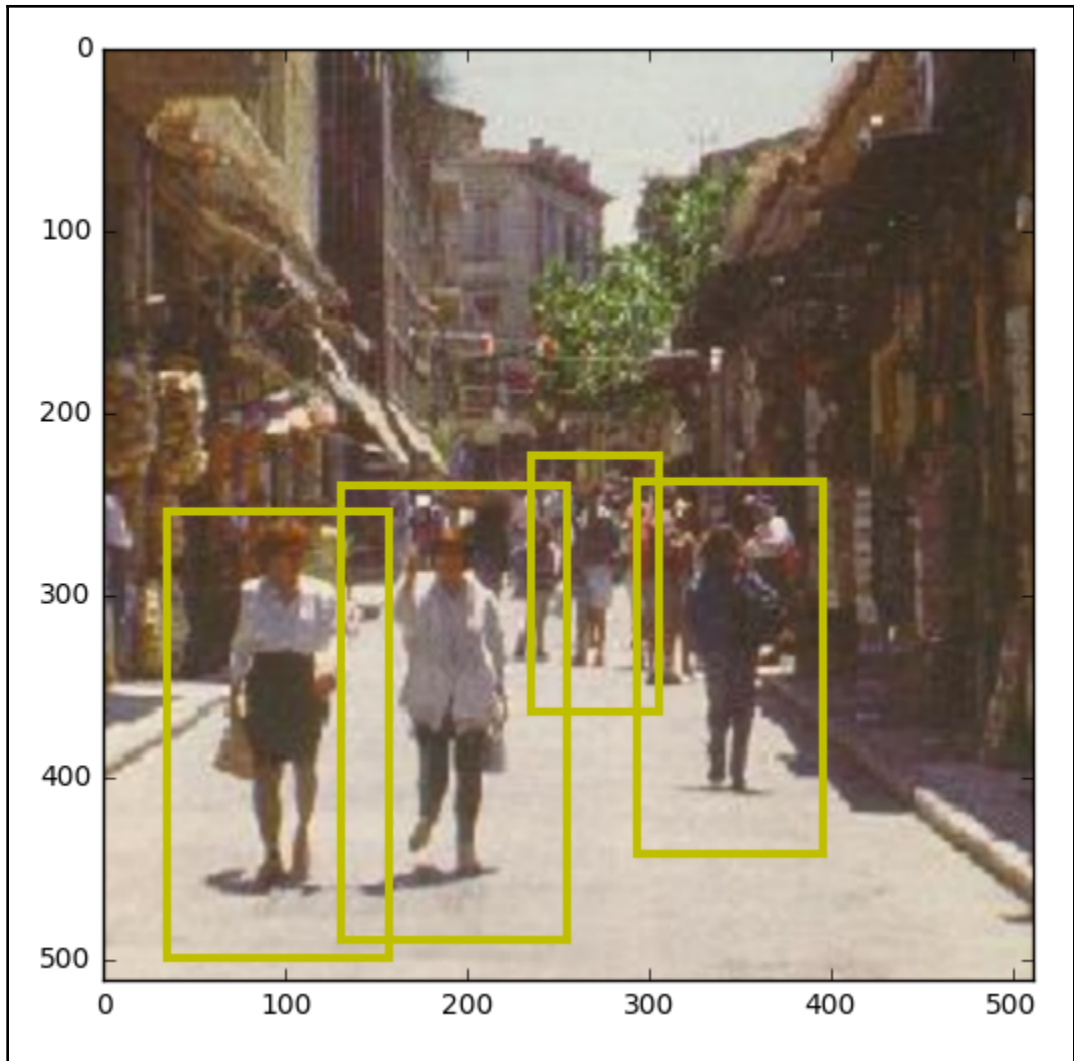


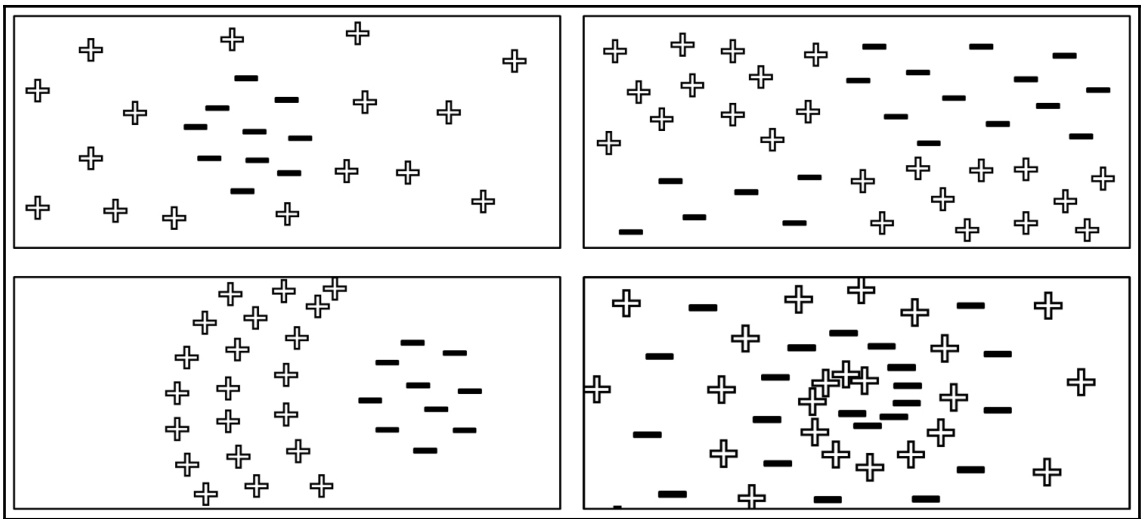






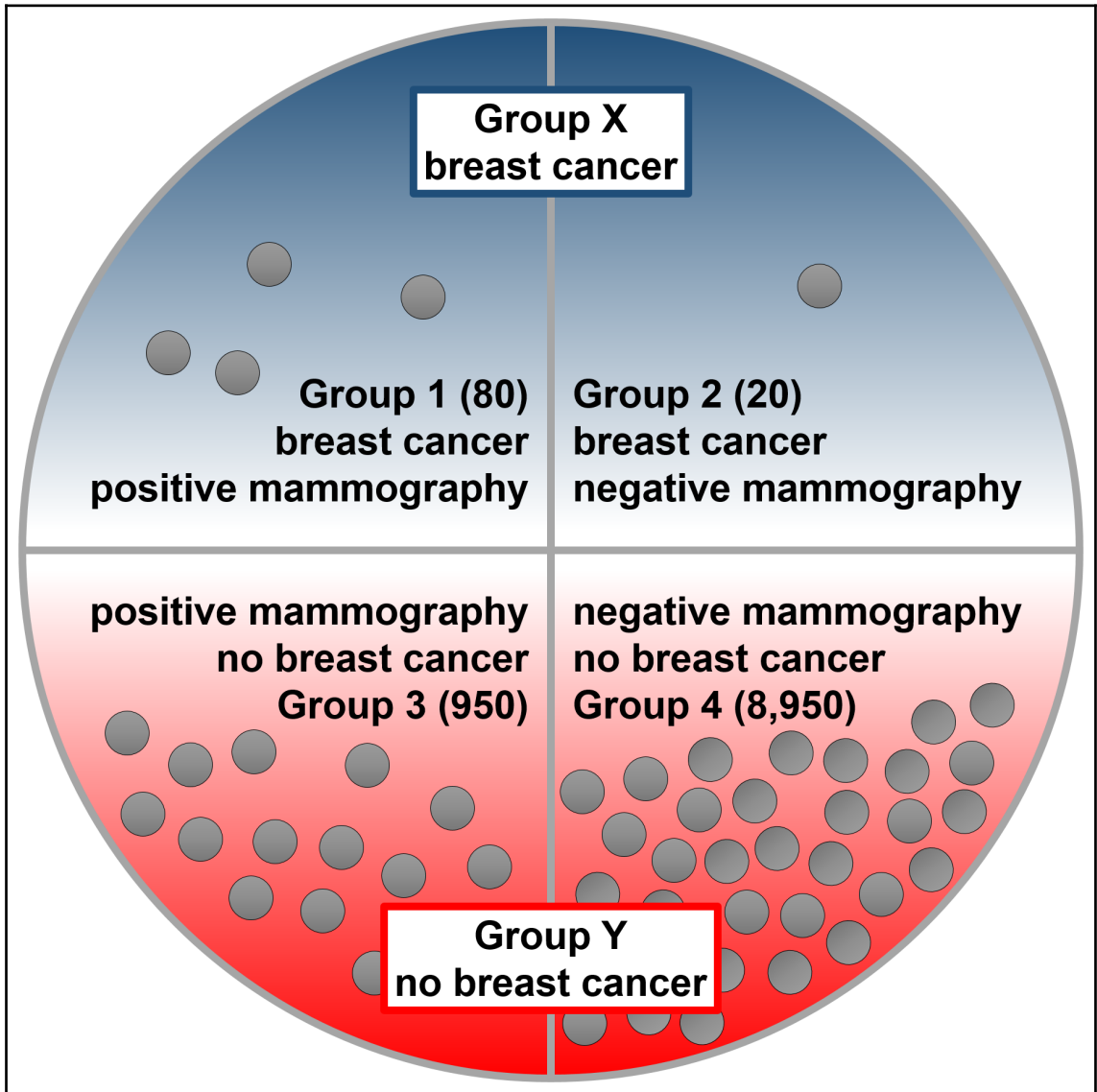


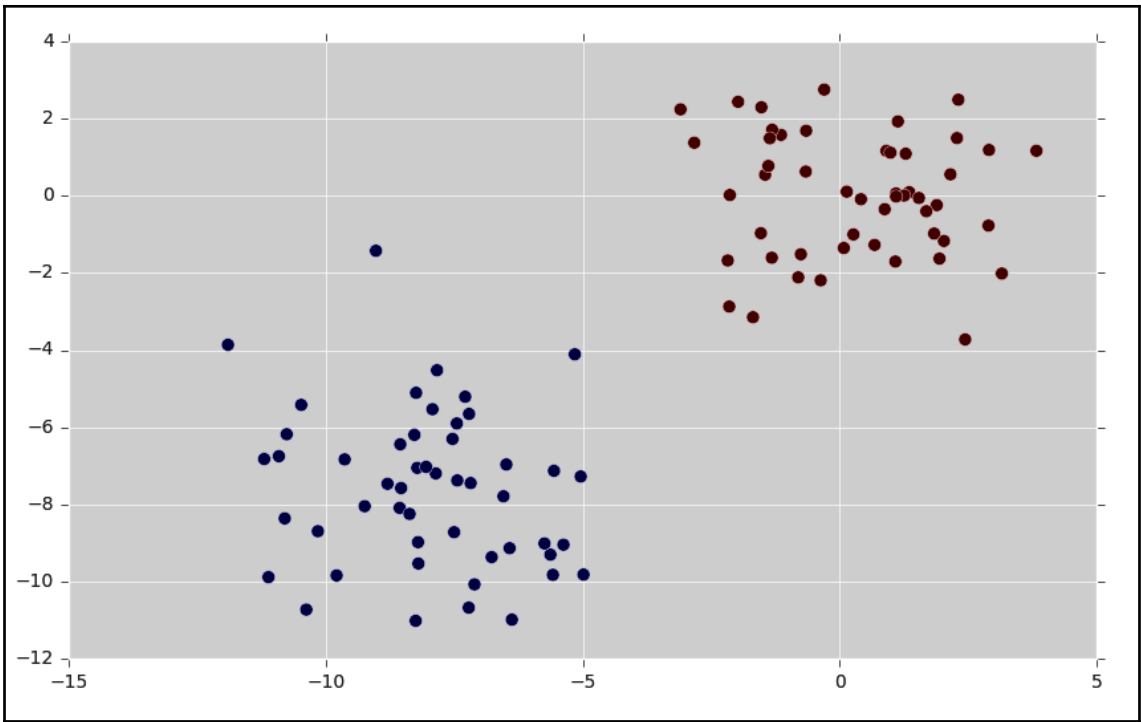


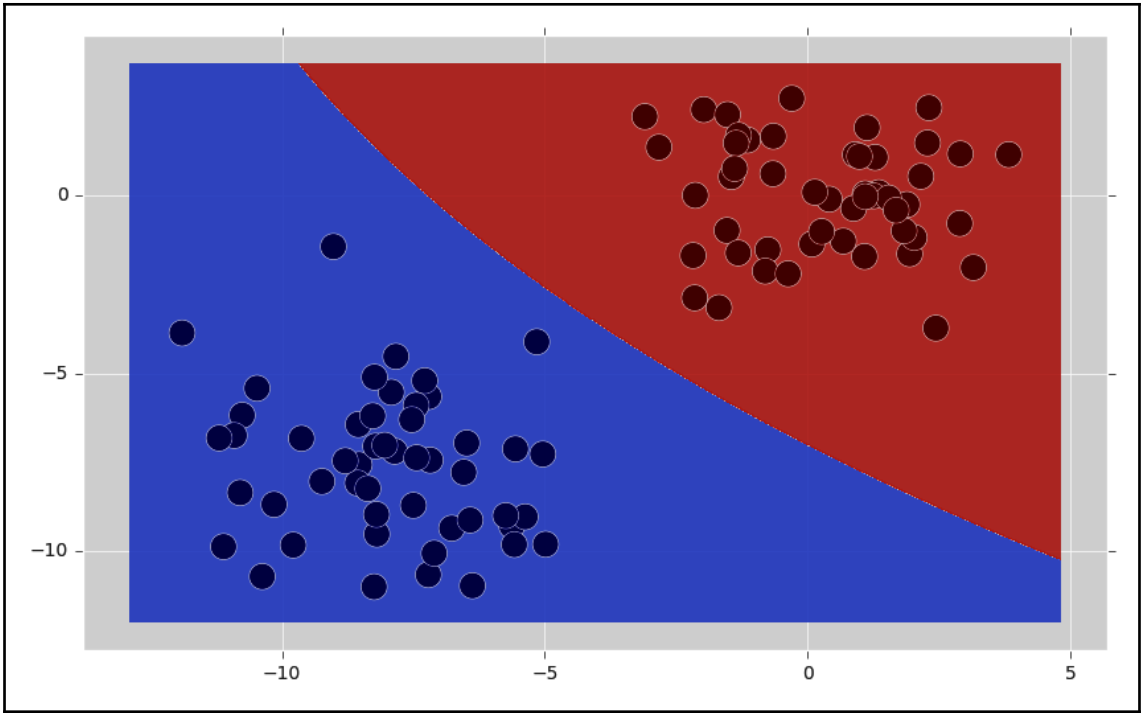


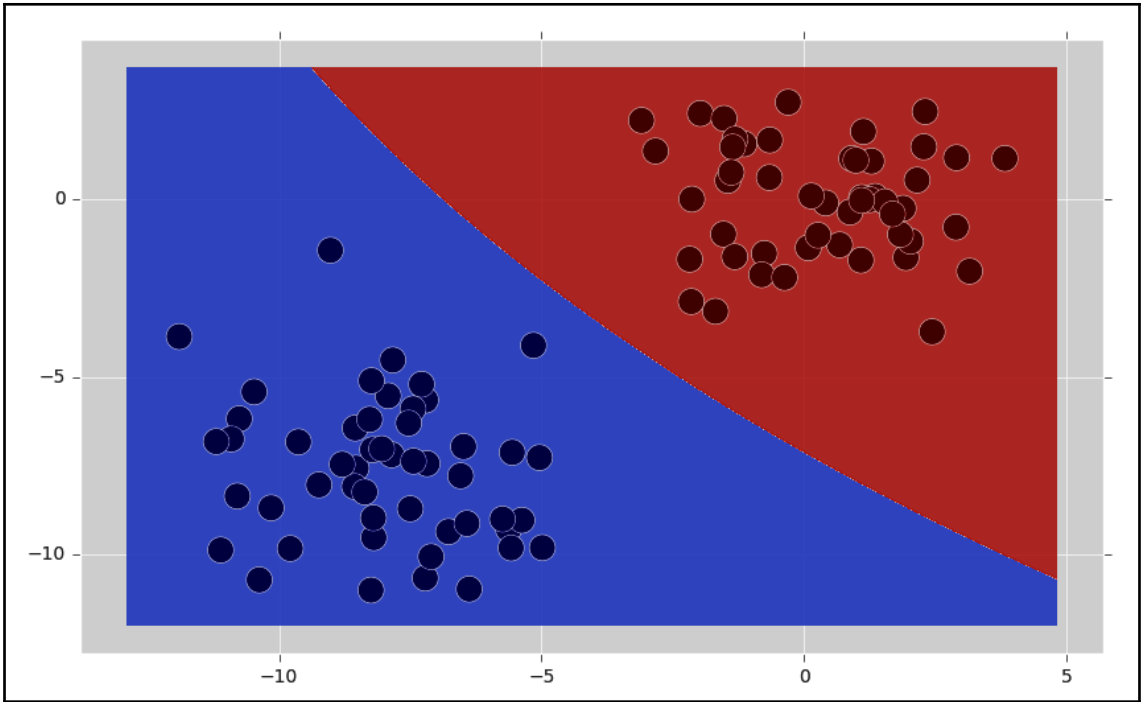
col_0	count
Activity	
LAYING	1407
SITTING	1286
STANDING	1374
WALKING	1226
WALKING_DOWNSTAIRS	986
WALKING_UPSTAIRS	1073

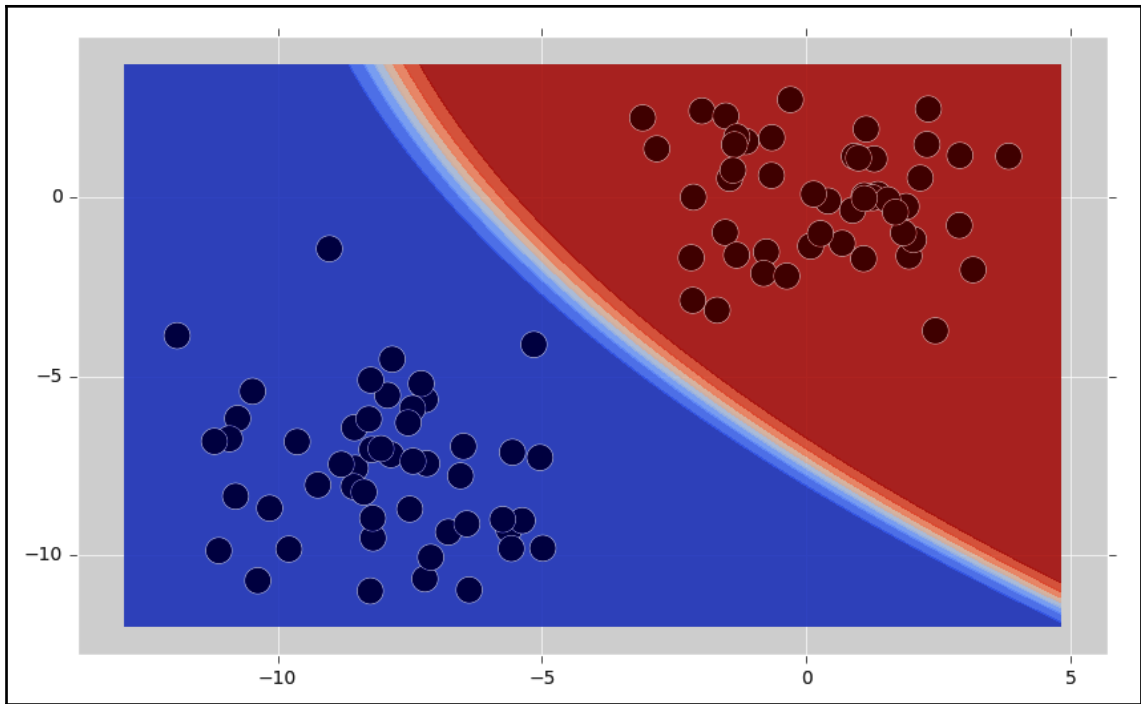
Chapter 7: Implementing a Spam Filter with Bayesian Learning







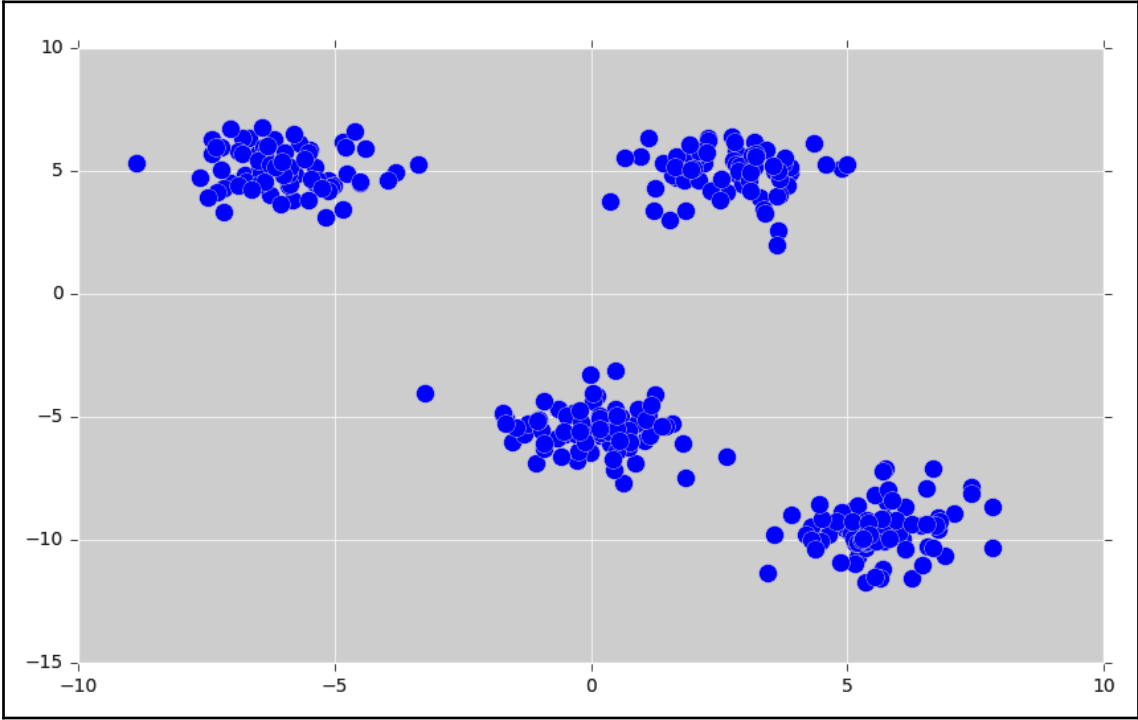


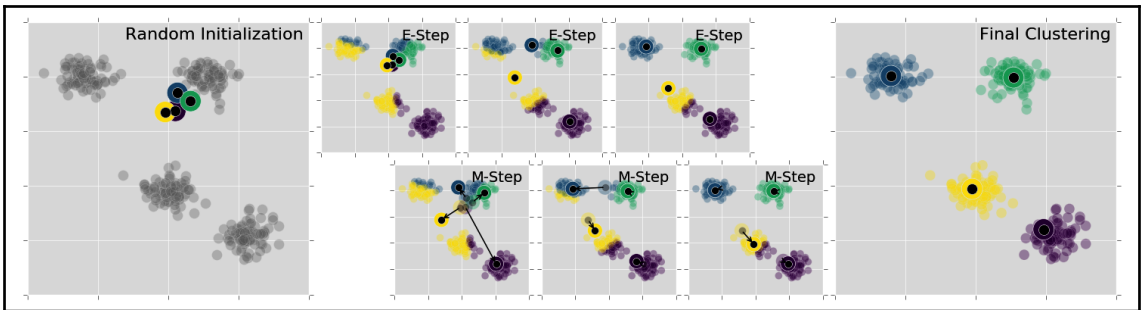
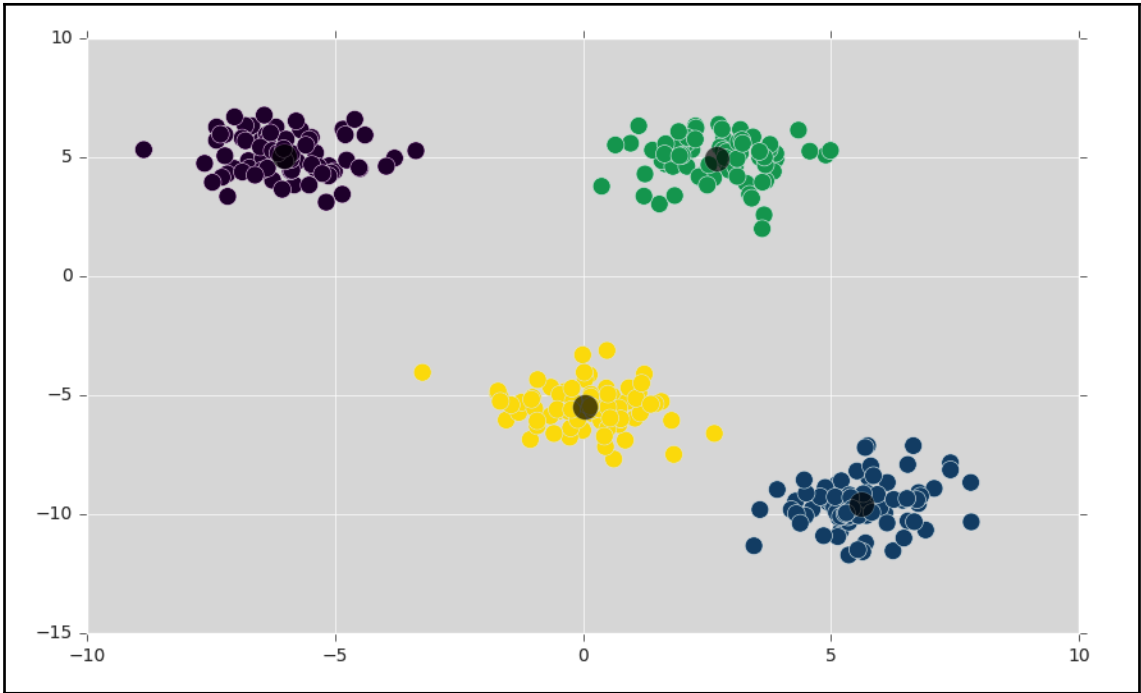


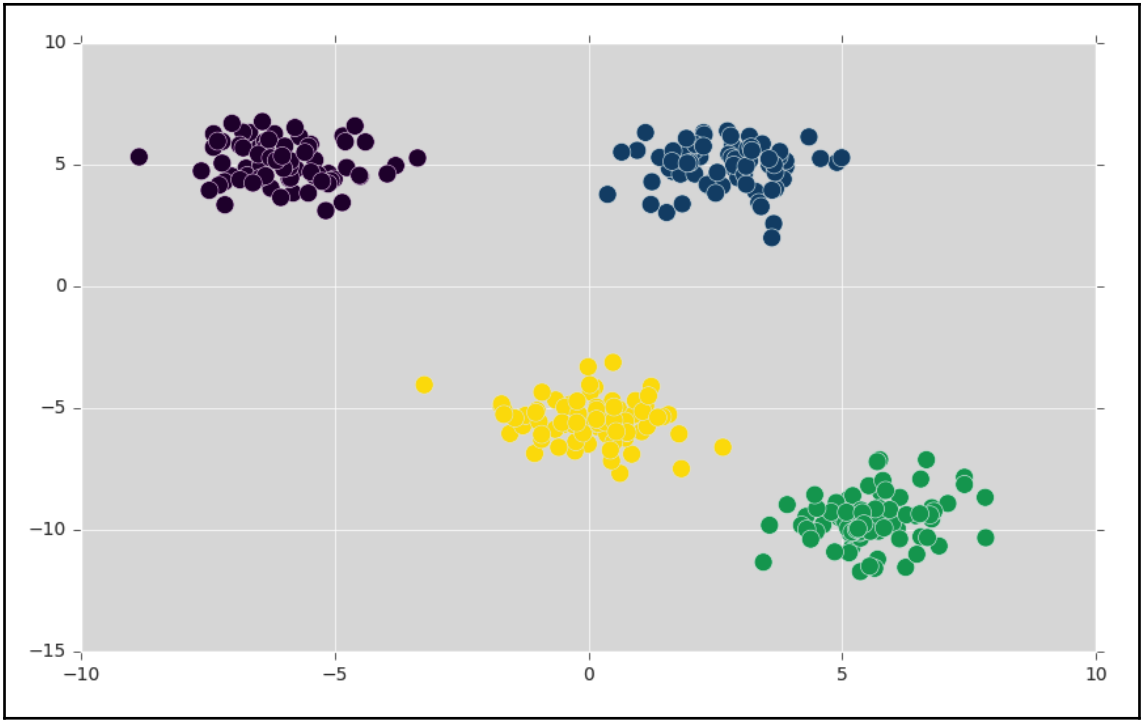
Out[7]:

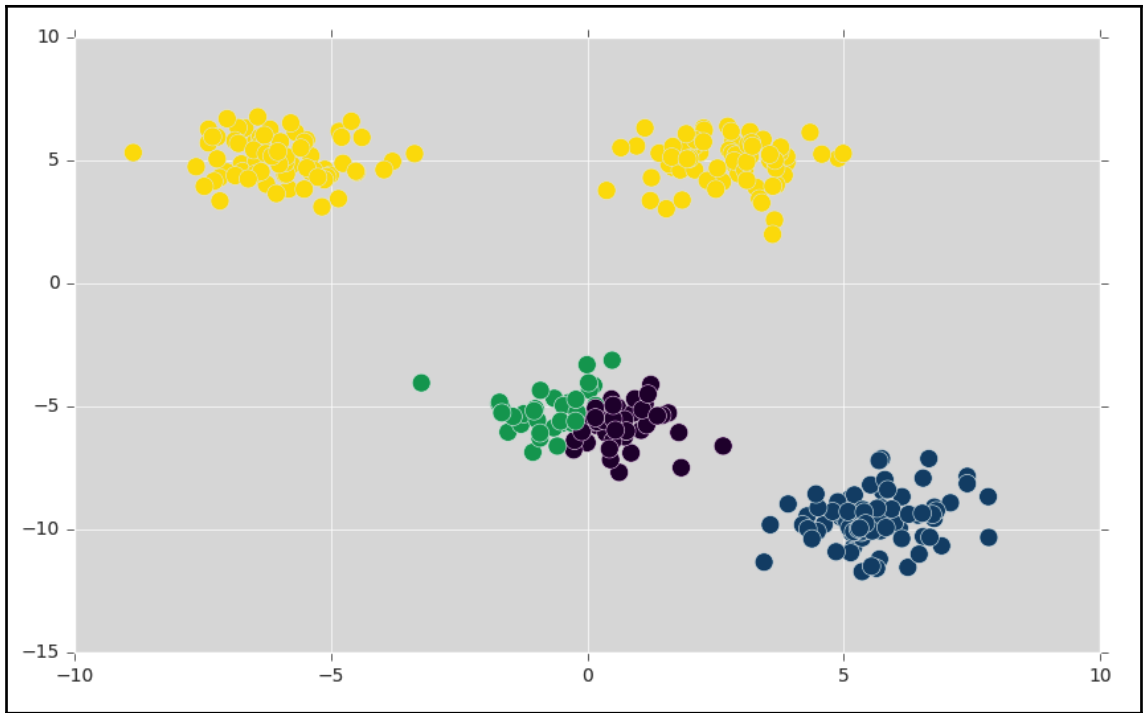
	class	model
0	cv2.ml.NormalBayesClassifier_create()	Normal Bayes
1	sklearn.naive_bayes.MultinomialNB()	Multinomial Bayes
2	sklearn.naive_bayes.BernoulliNB()	Bernoulli Bayes

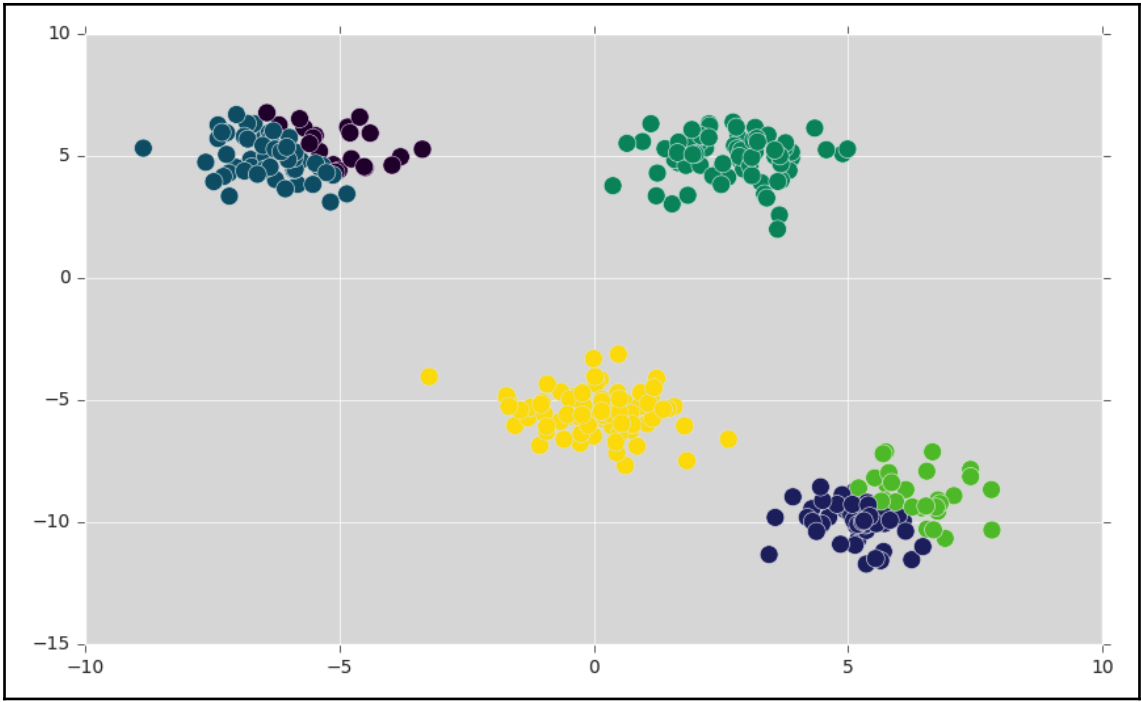
Chapter 8: Discovering Hidden Structures with Unsupervised Learning

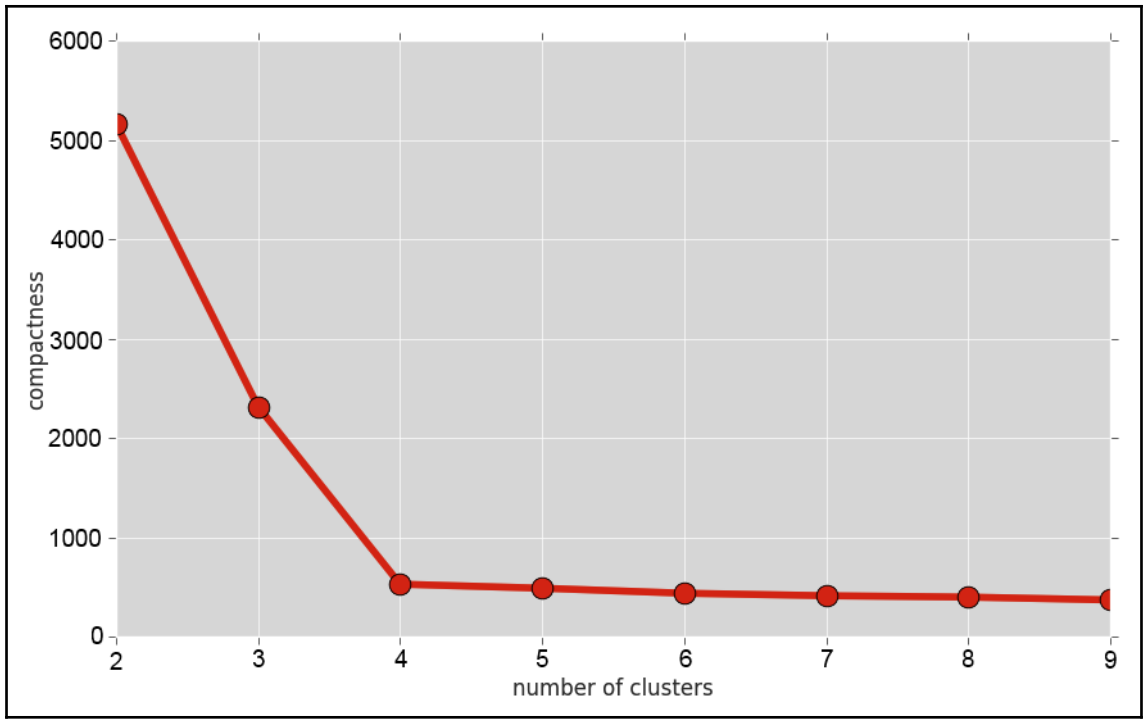


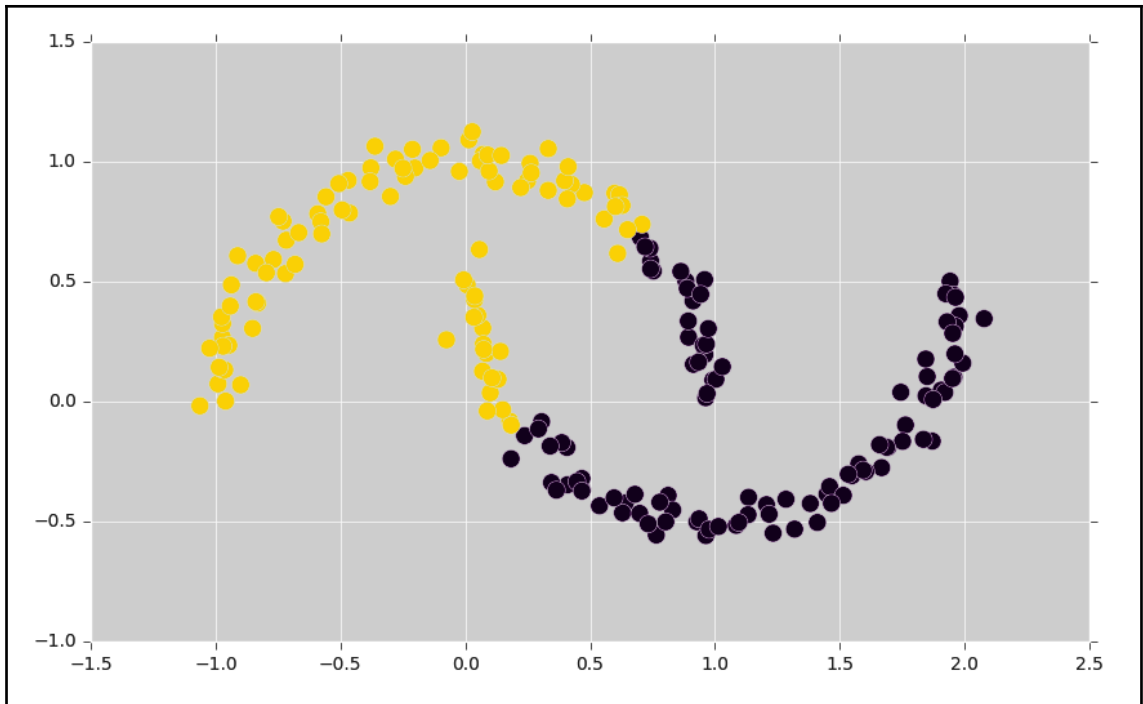


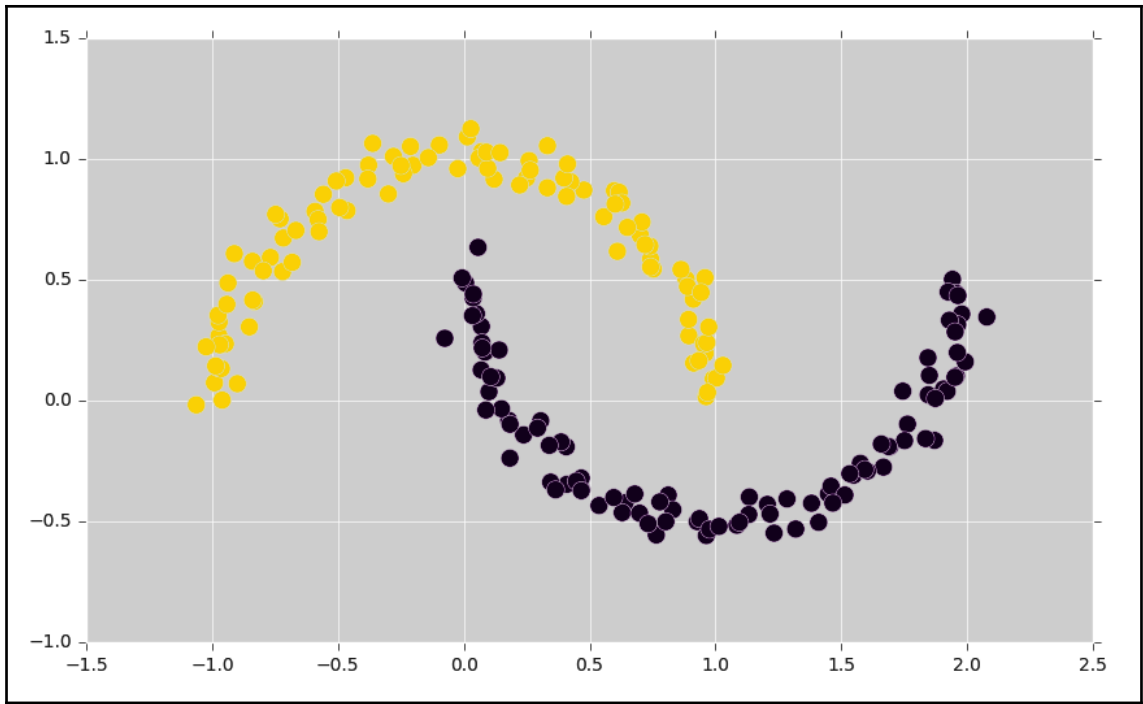


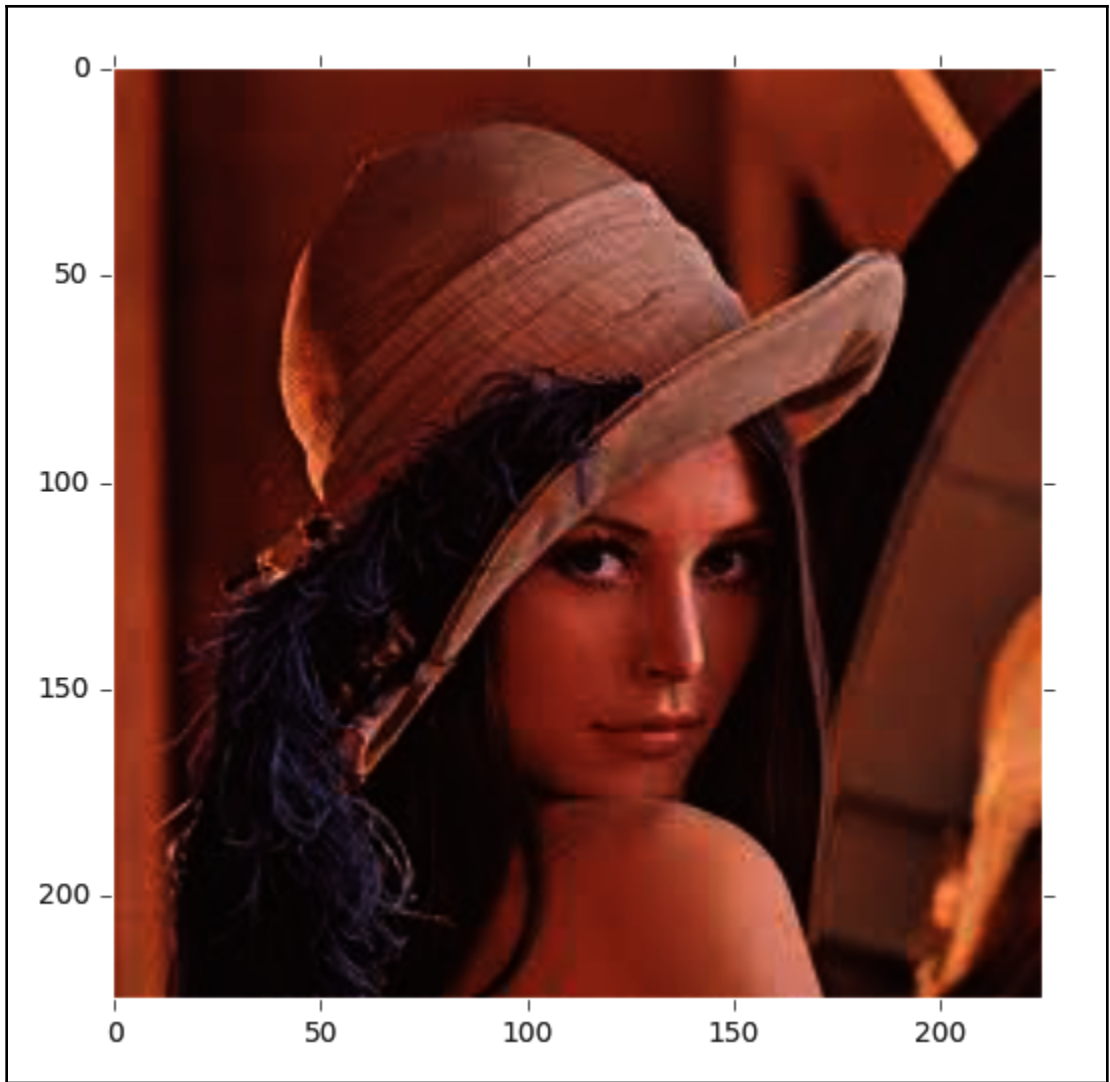


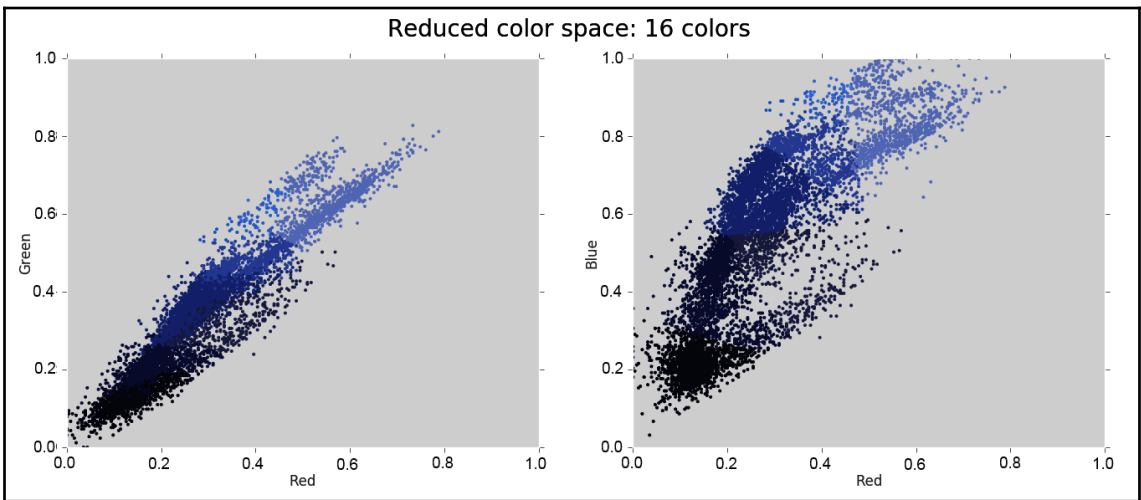
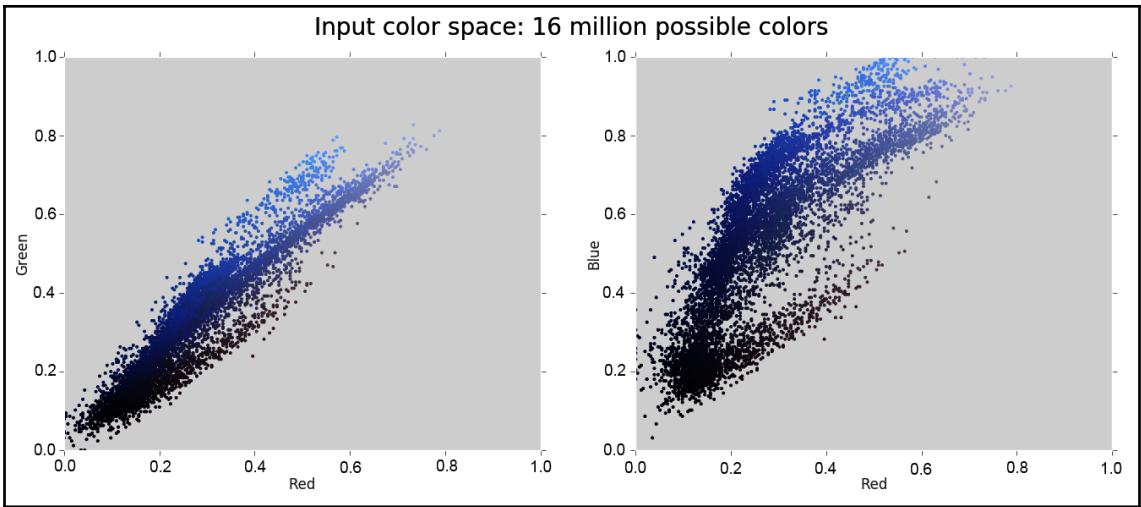


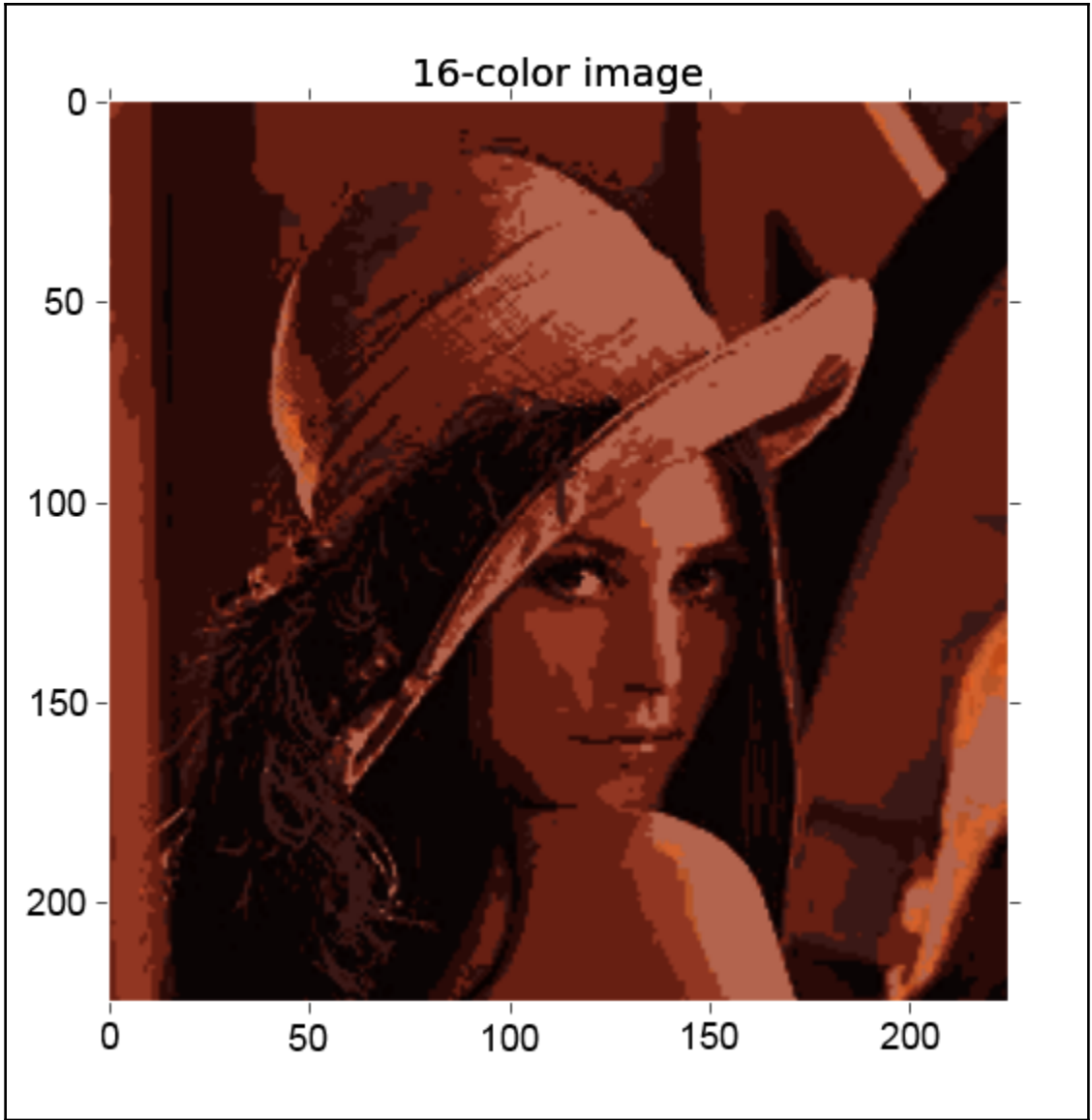


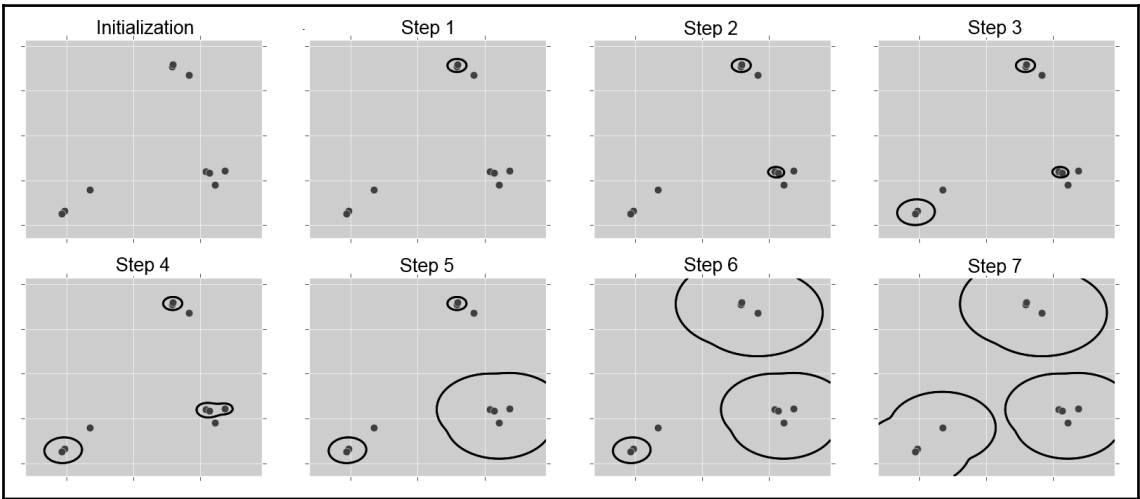
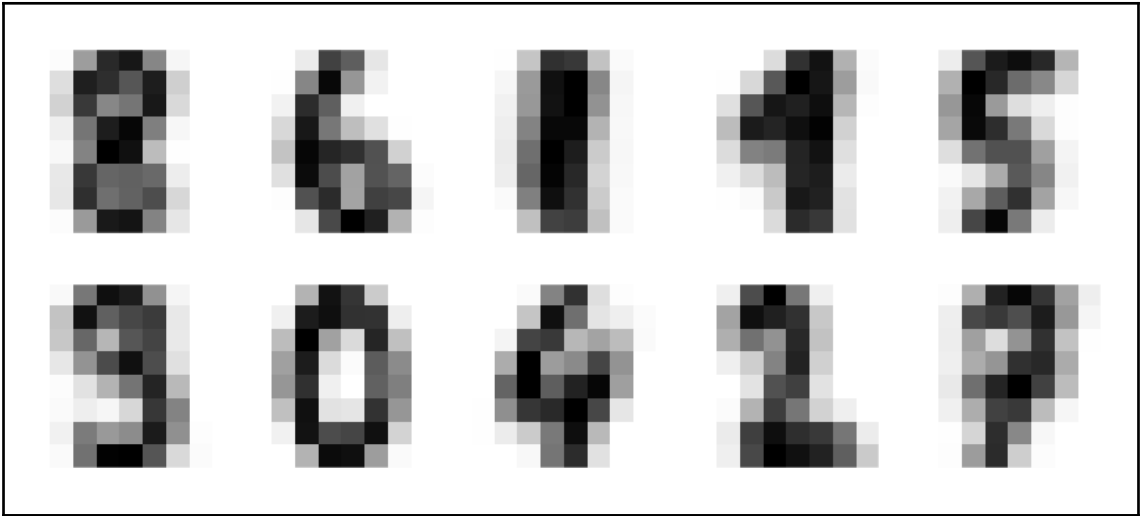


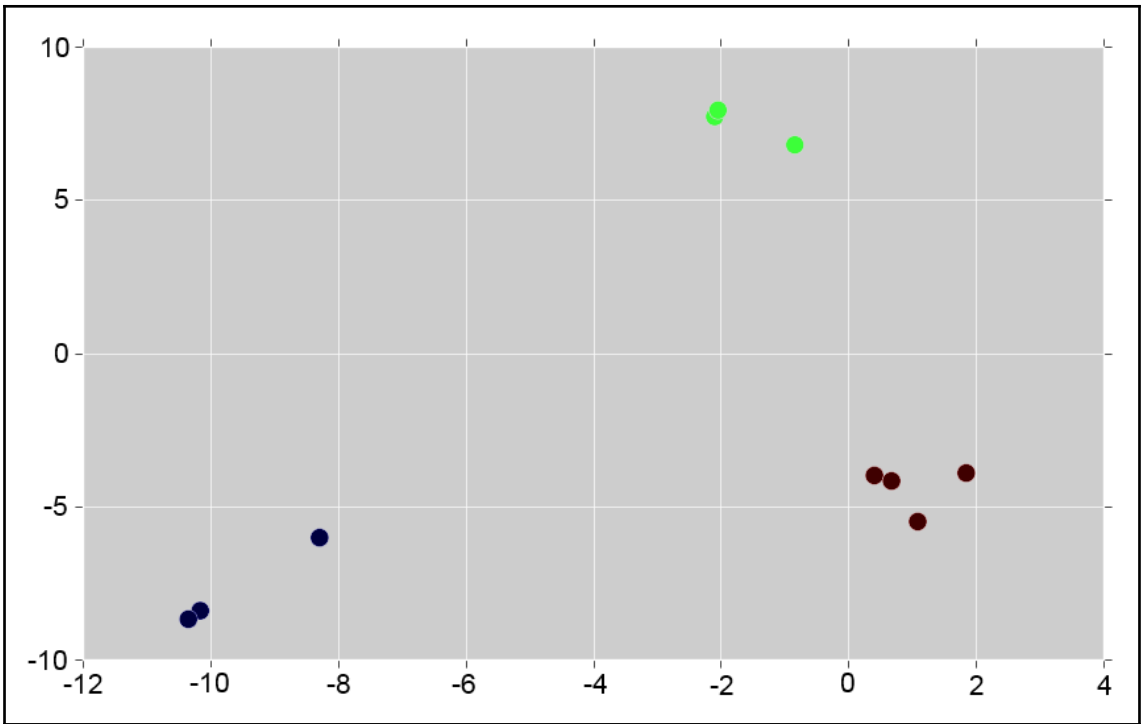




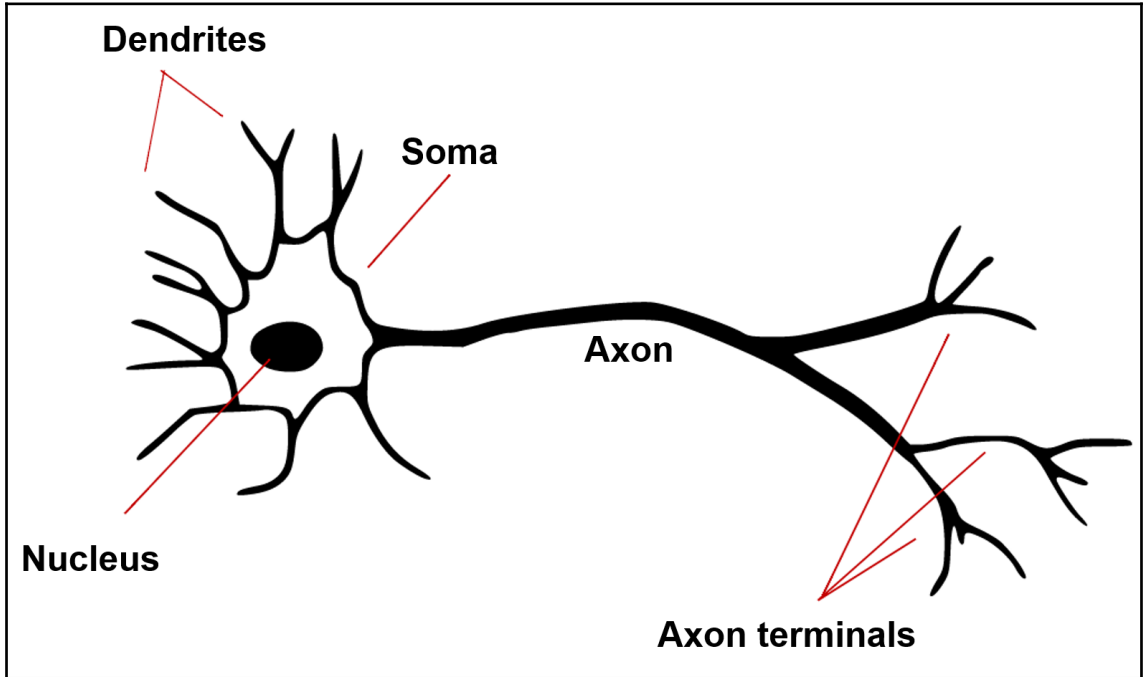


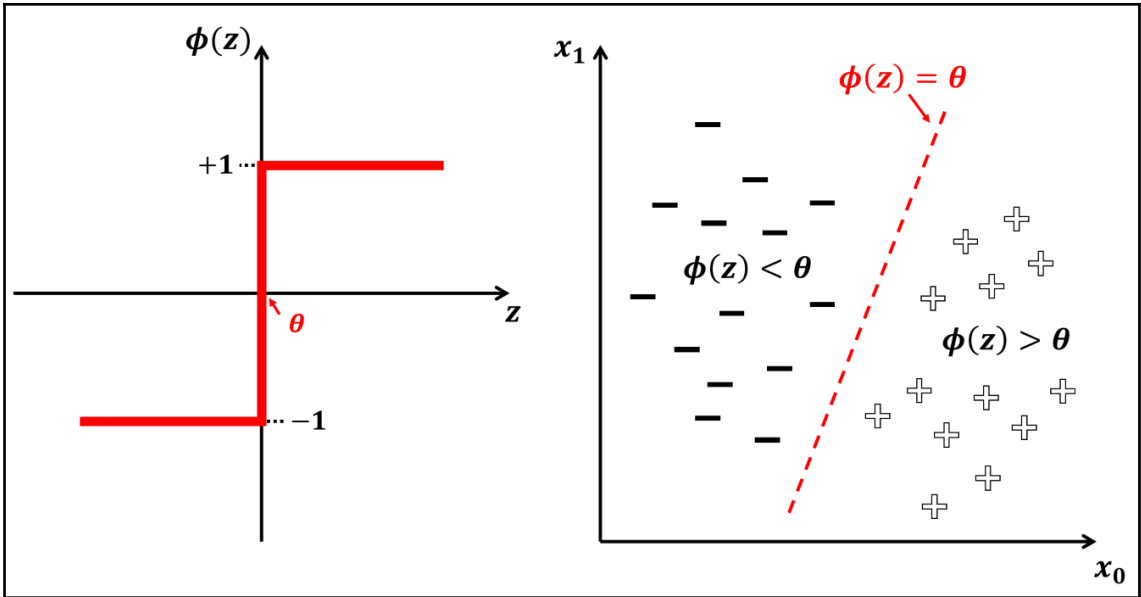


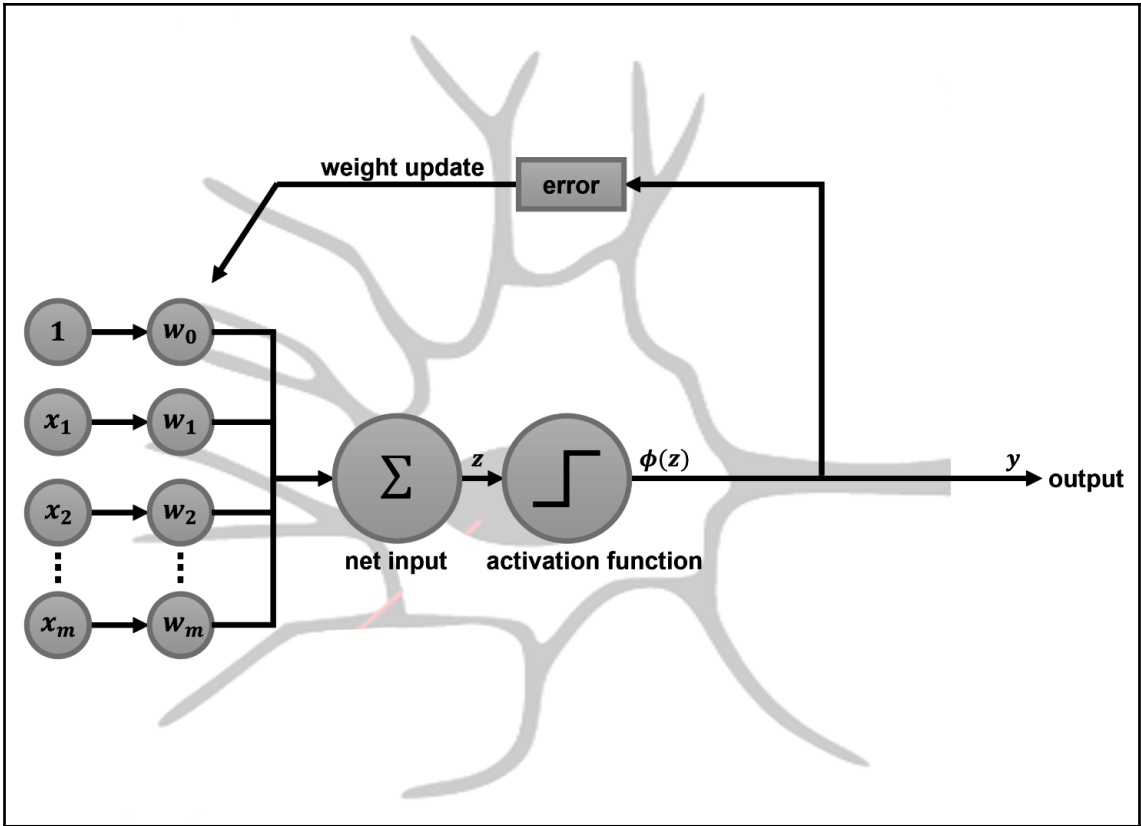


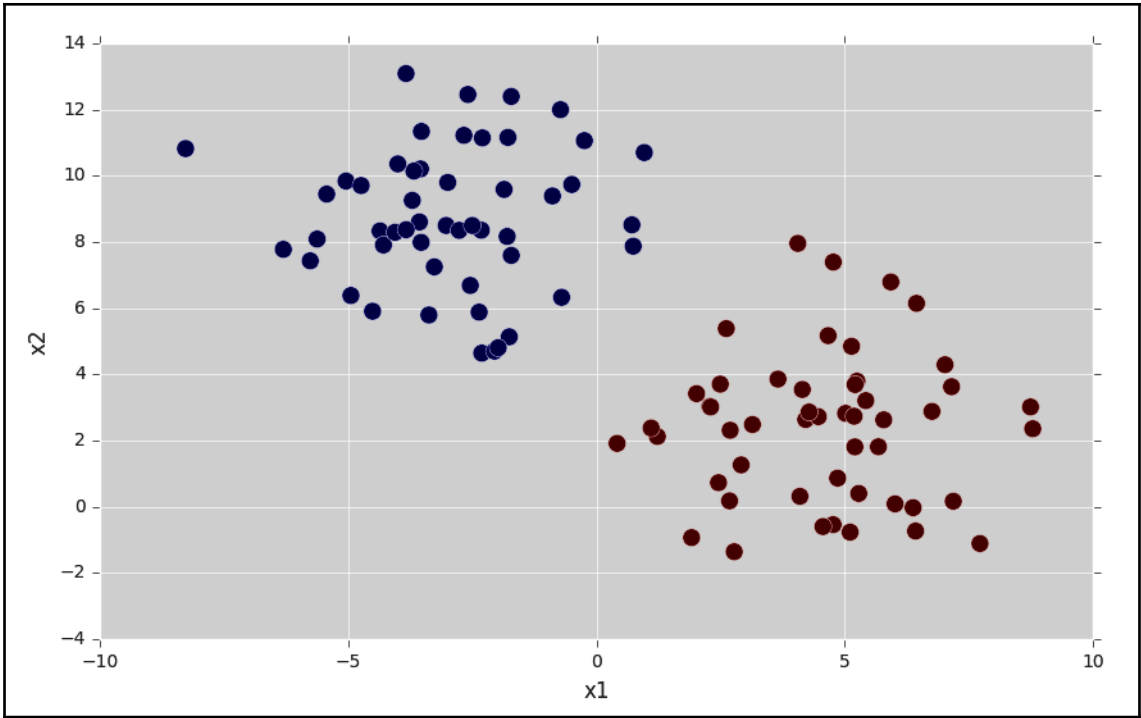


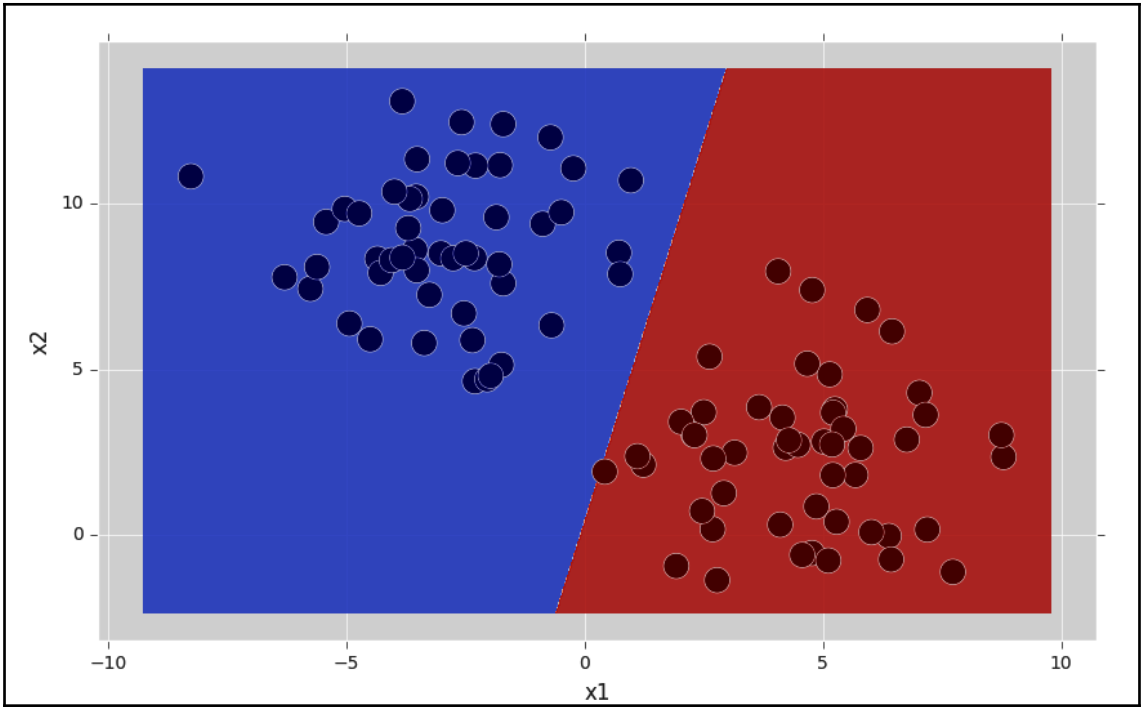
Chapter 9: Using Deep Learning to Classify Handwritten Digits

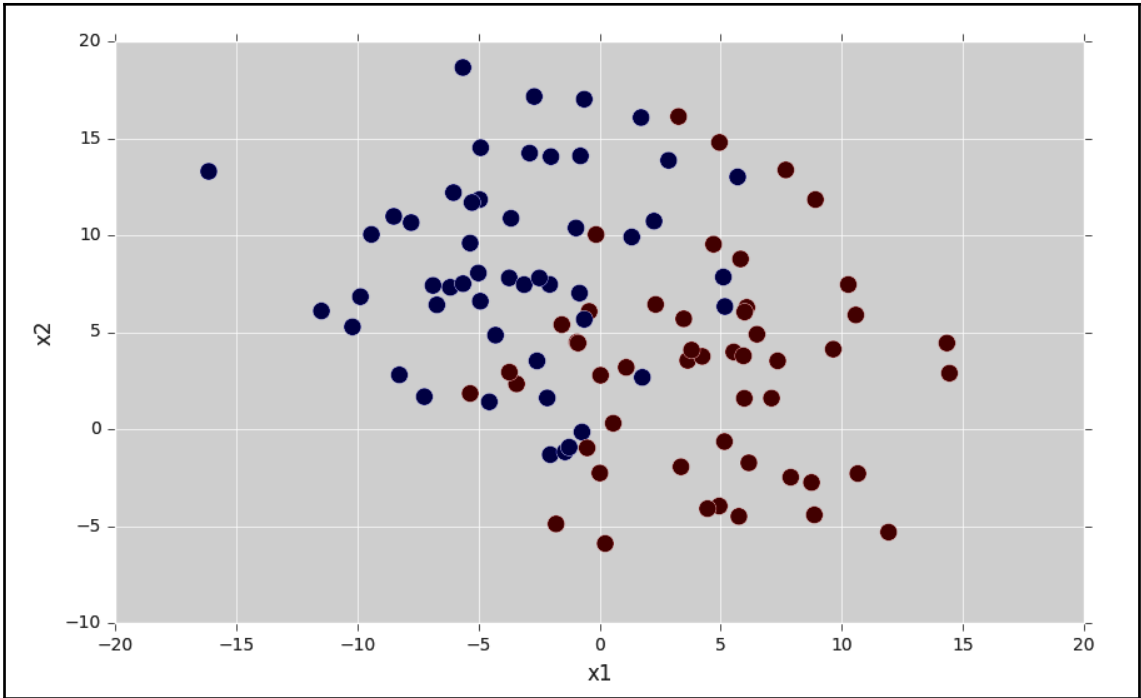


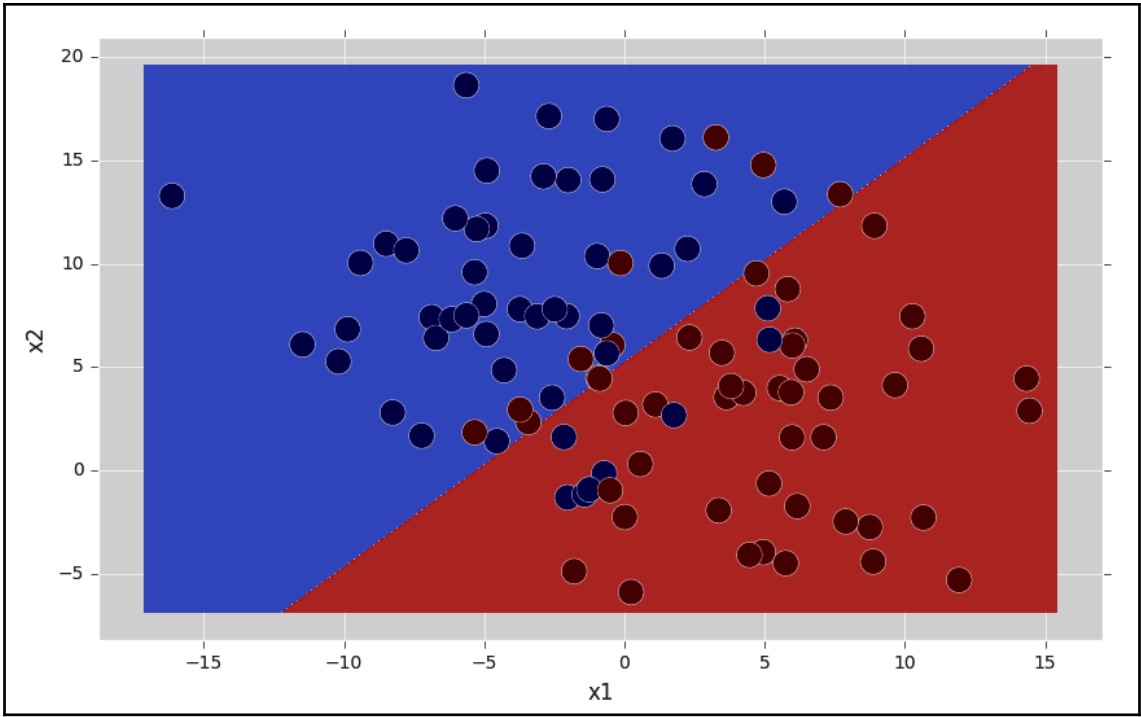


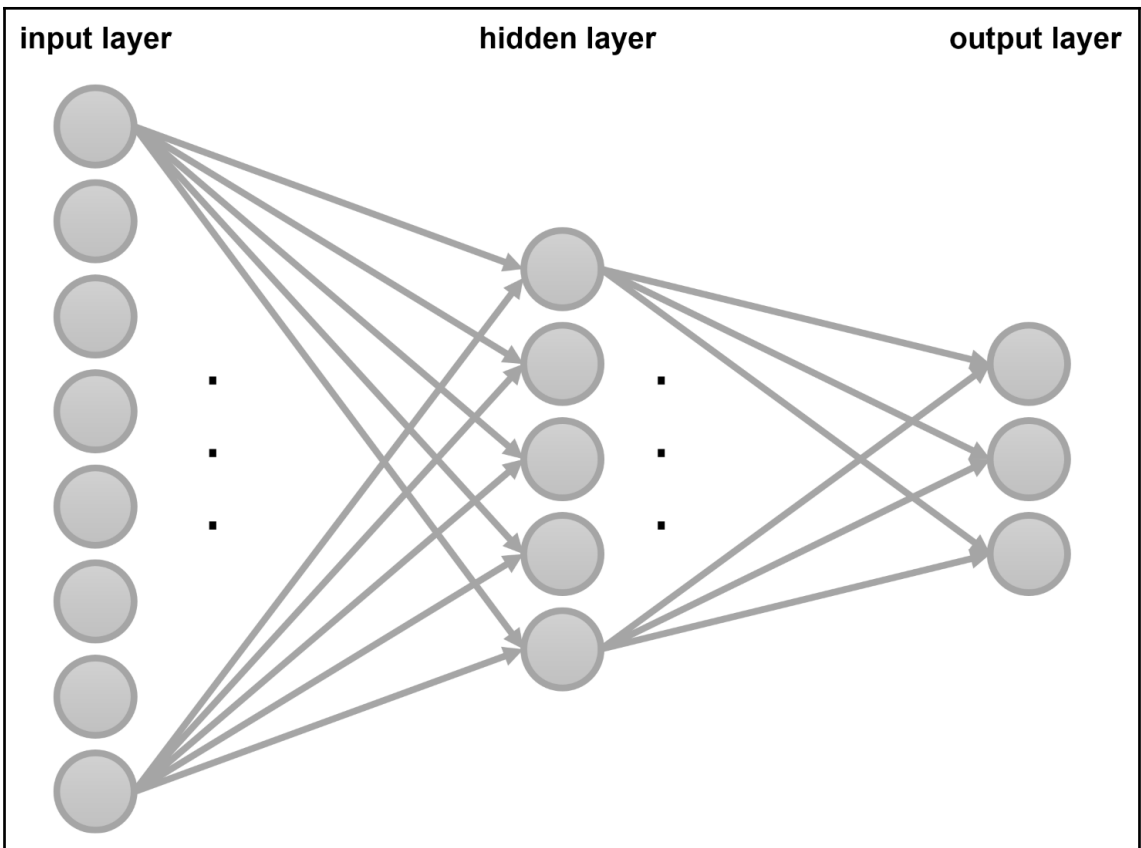


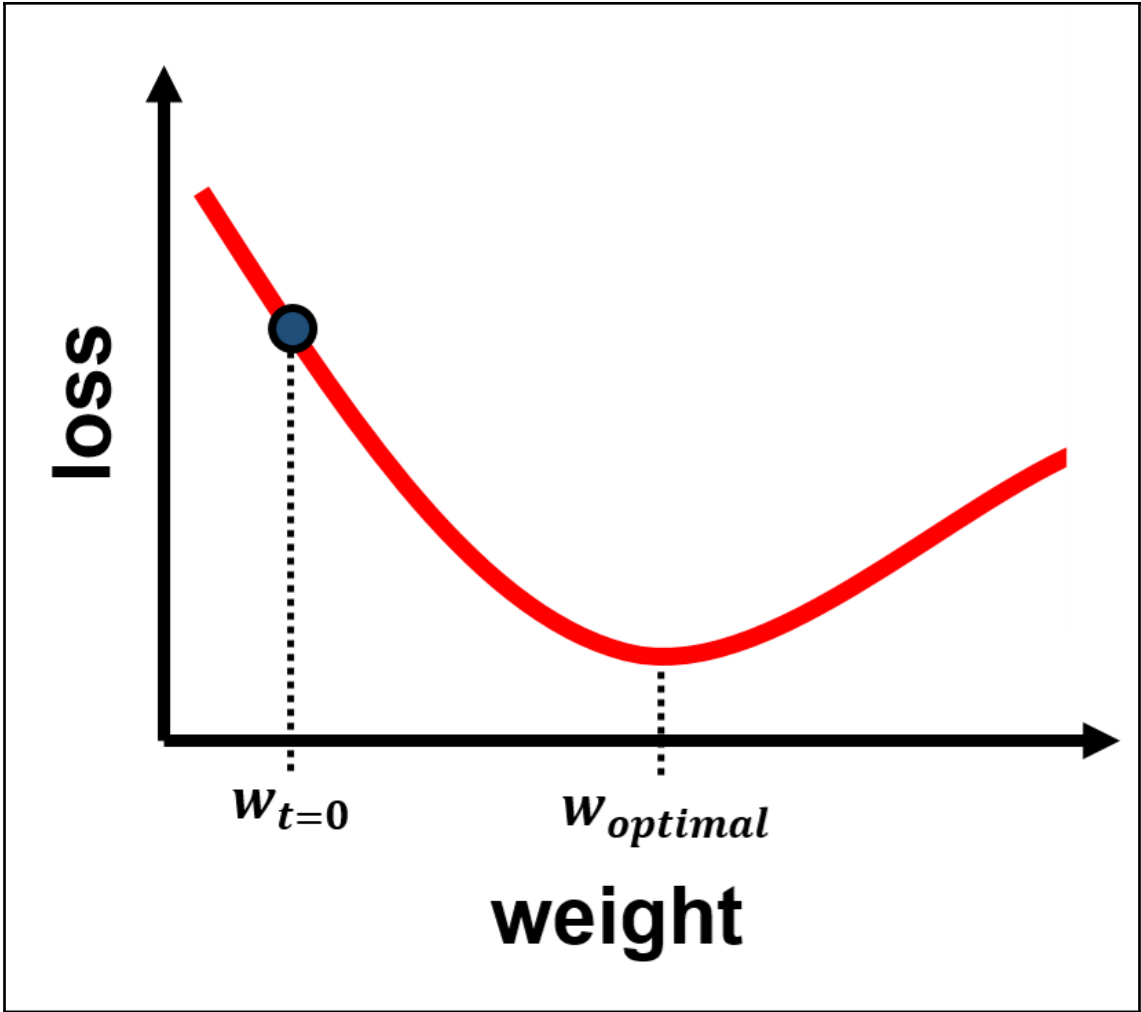


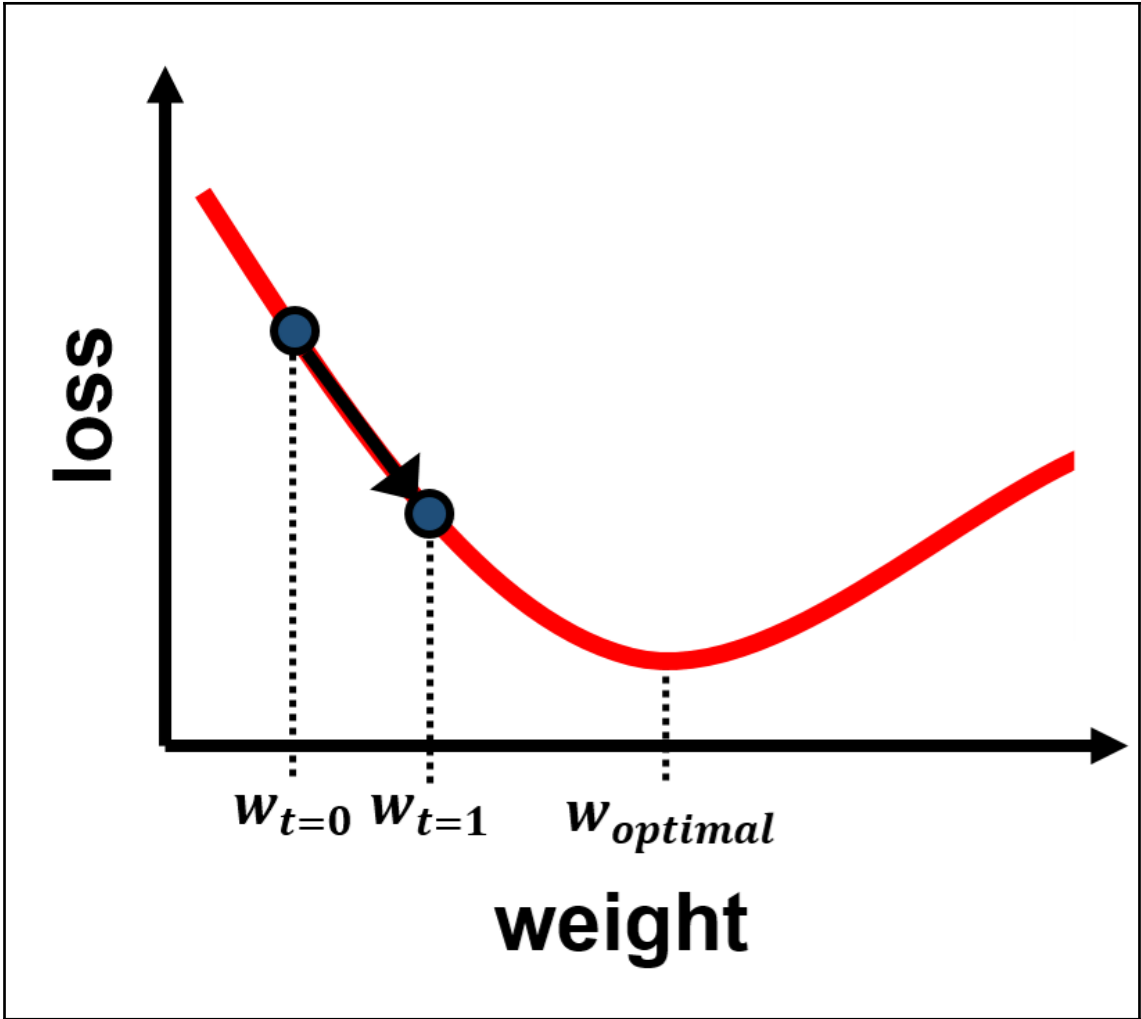


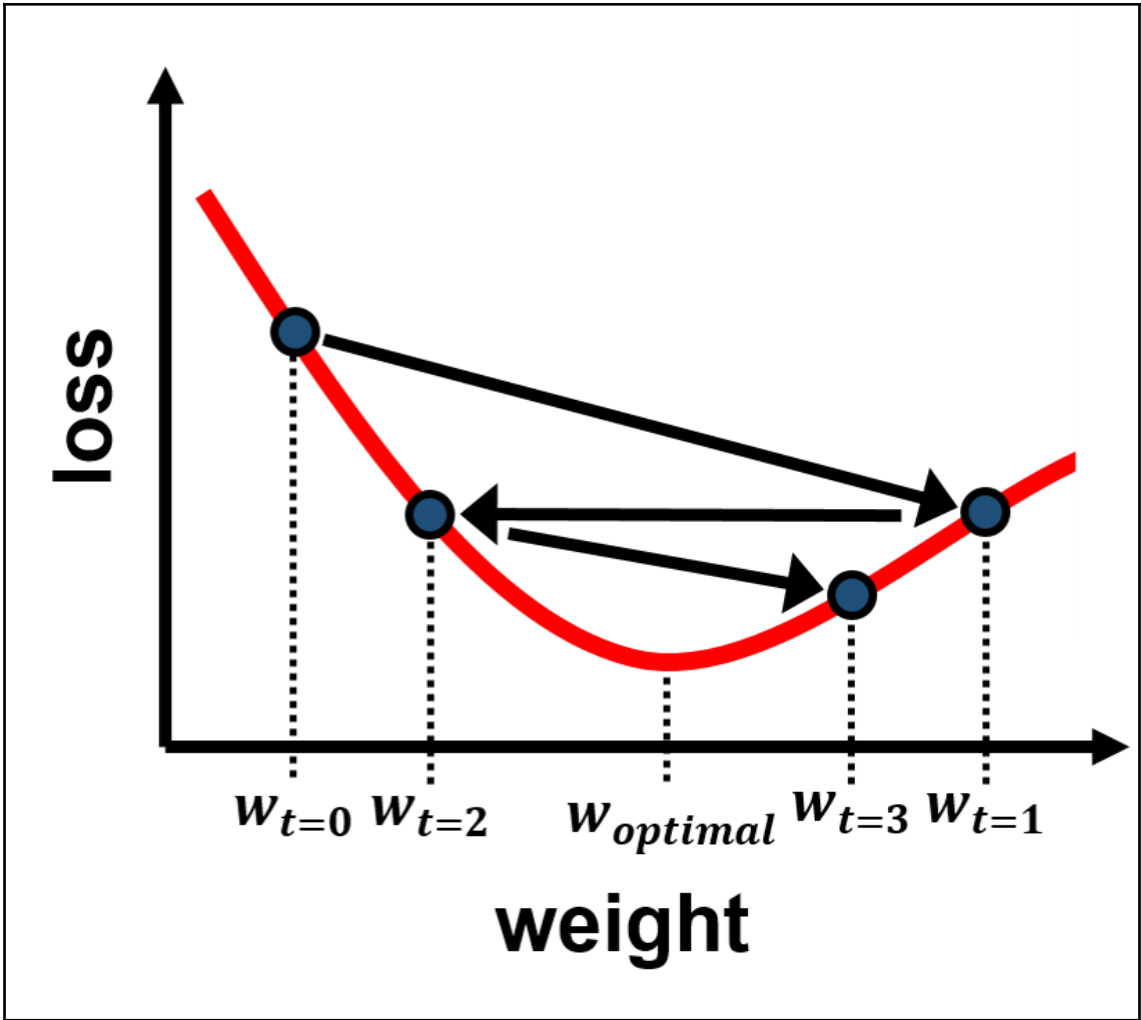


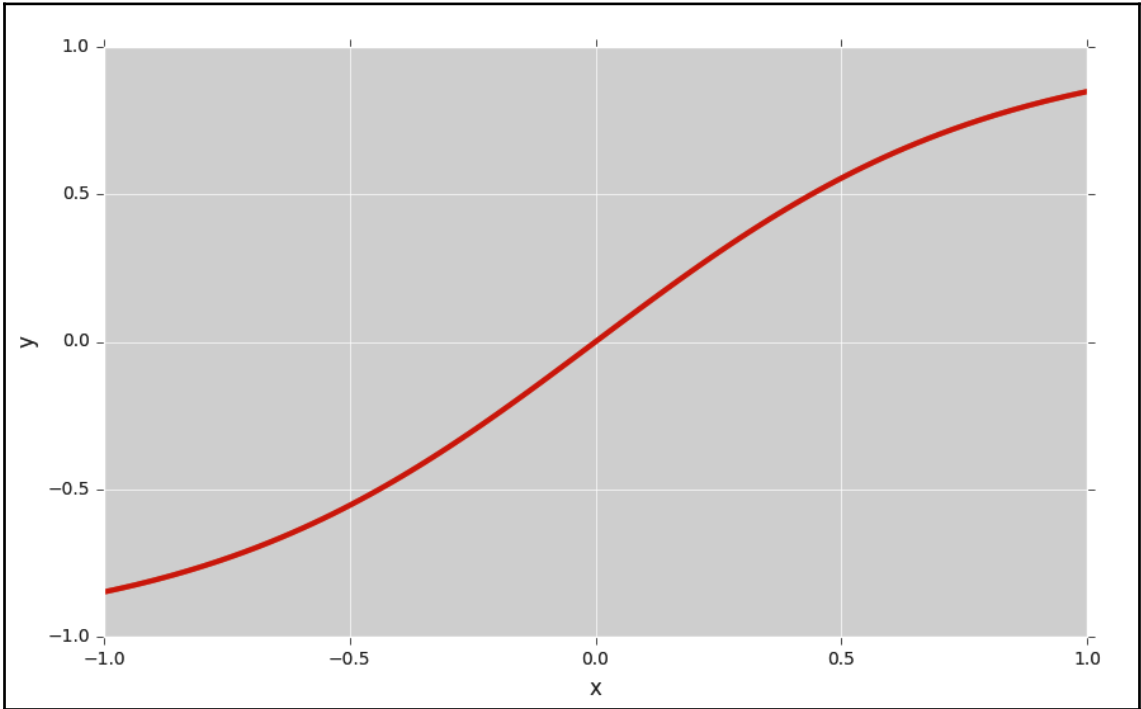
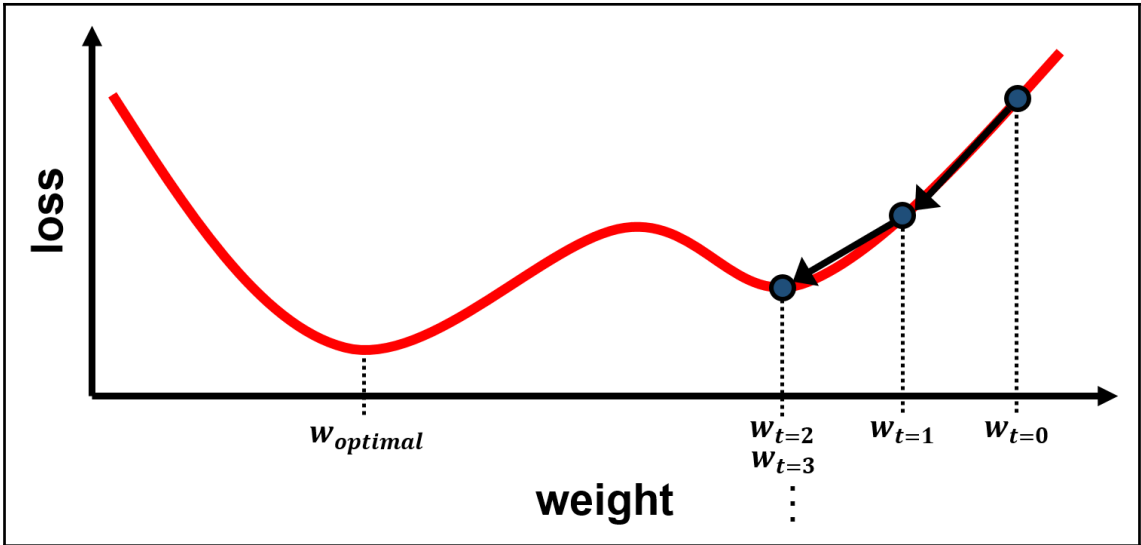


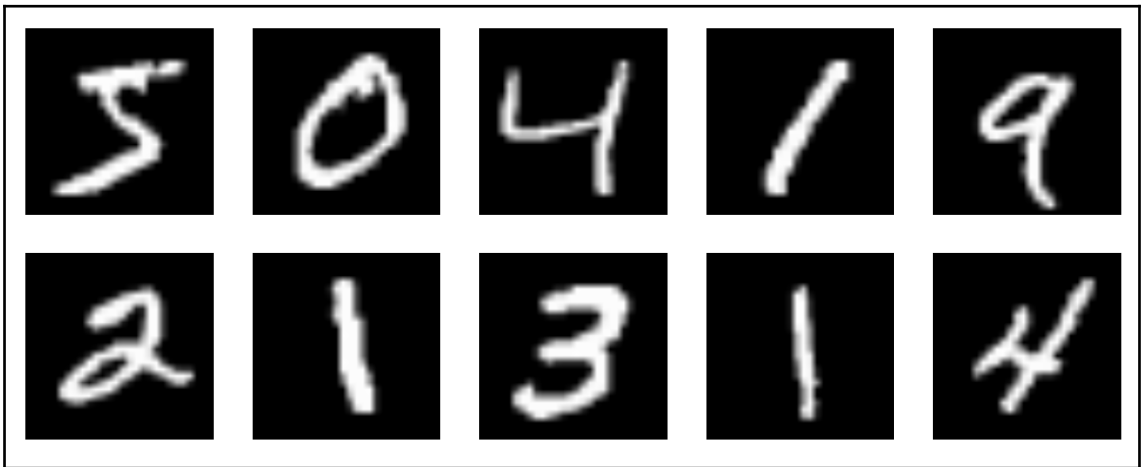
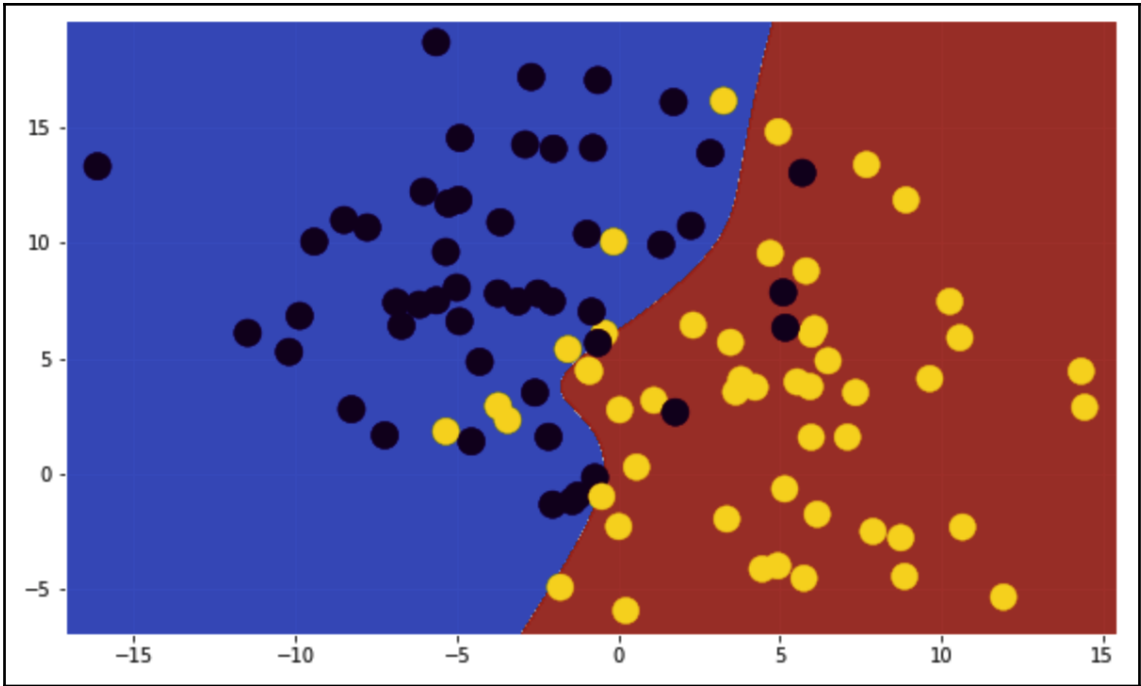






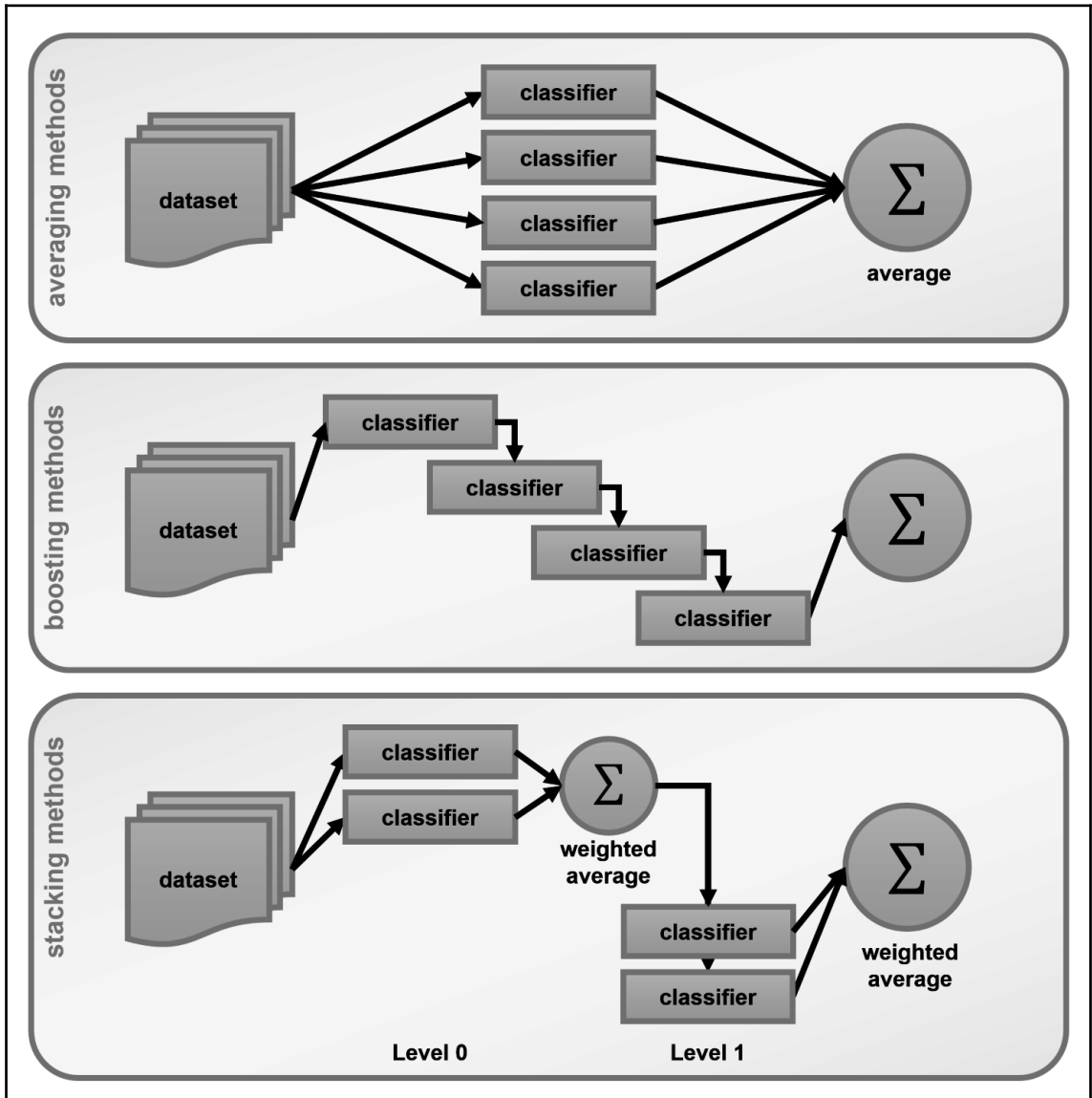


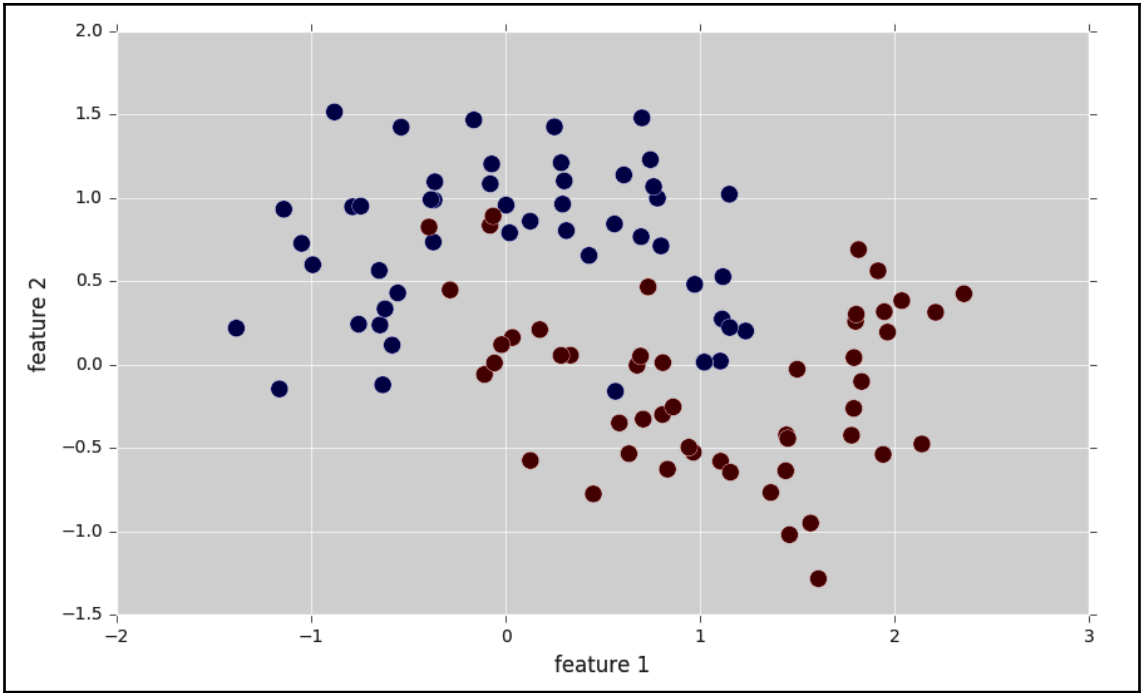


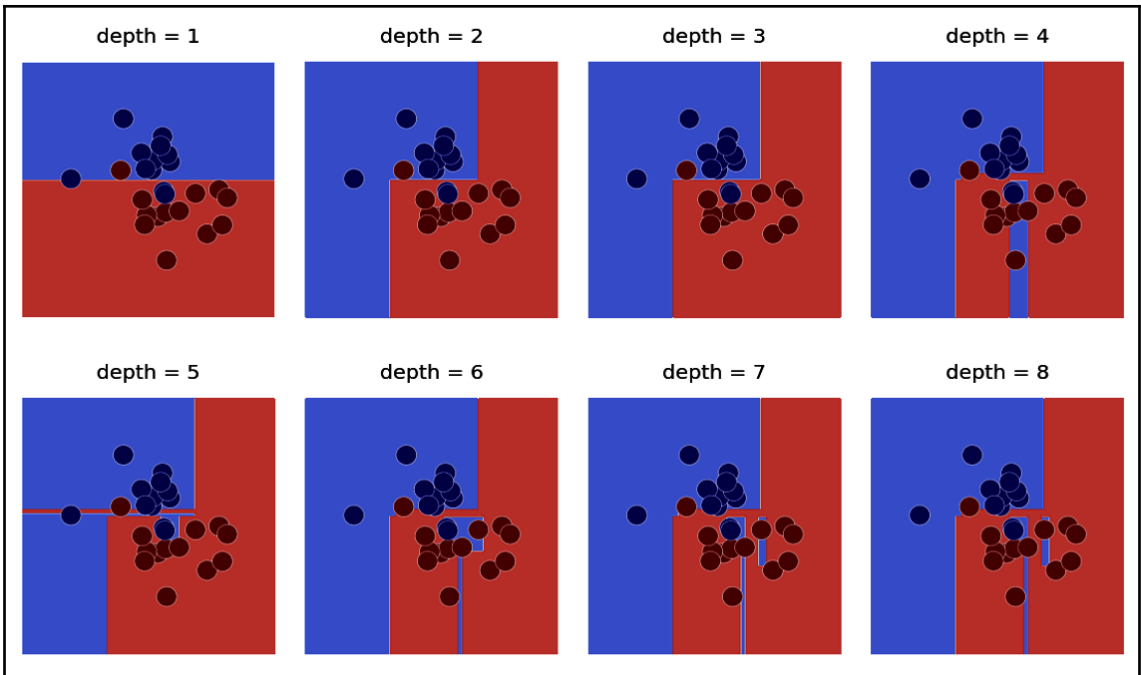


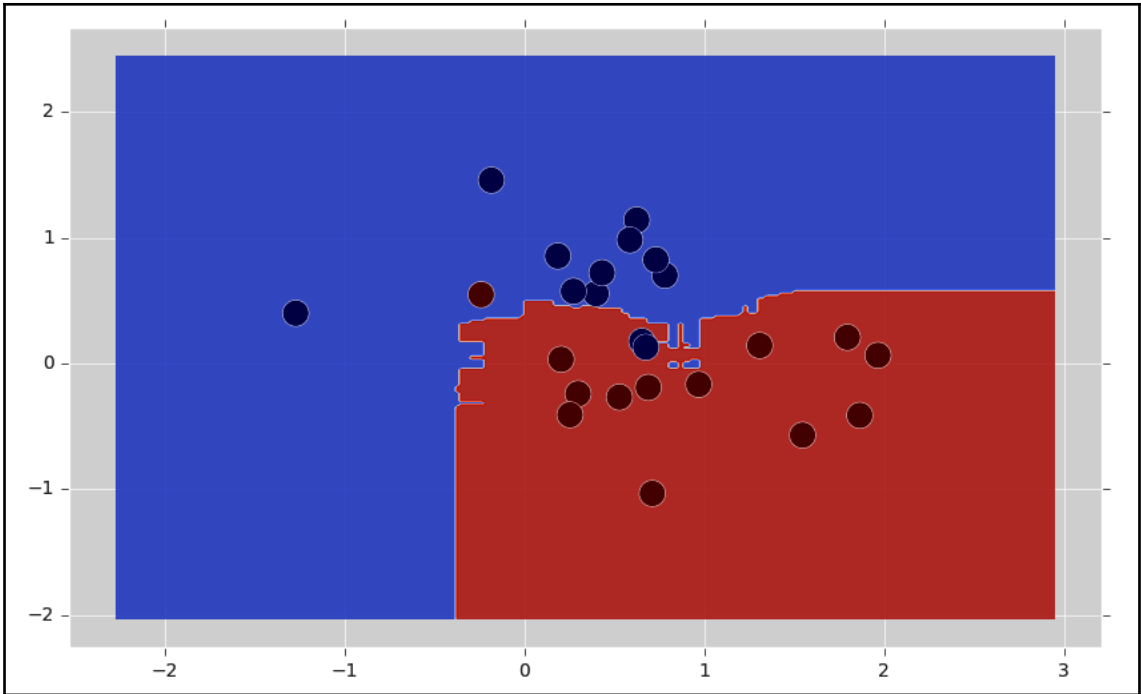
Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 26, 26, 32)	320
activation_5 (Activation)	(None, 26, 26, 32)	0
conv2d_5 (Conv2D)	(None, 24, 24, 32)	9248
activation_6 (Activation)	(None, 24, 24, 32)	0
max_pooling2d_3 (MaxPooling2D)	(None, 12, 12, 32)	0
dropout_3 (Dropout)	(None, 12, 12, 32)	0
flatten_3 (Flatten)	(None, 4608)	0
dense_3 (Dense)	(None, 128)	589952
activation_7 (Activation)	(None, 128)	0
dropout_4 (Dropout)	(None, 128)	0
dense_4 (Dense)	(None, 10)	1290
activation_8 (Activation)	(None, 10)	0
=====		
Total params: 600,810		
Trainable params: 600,810		
Non-trainable params: 0		

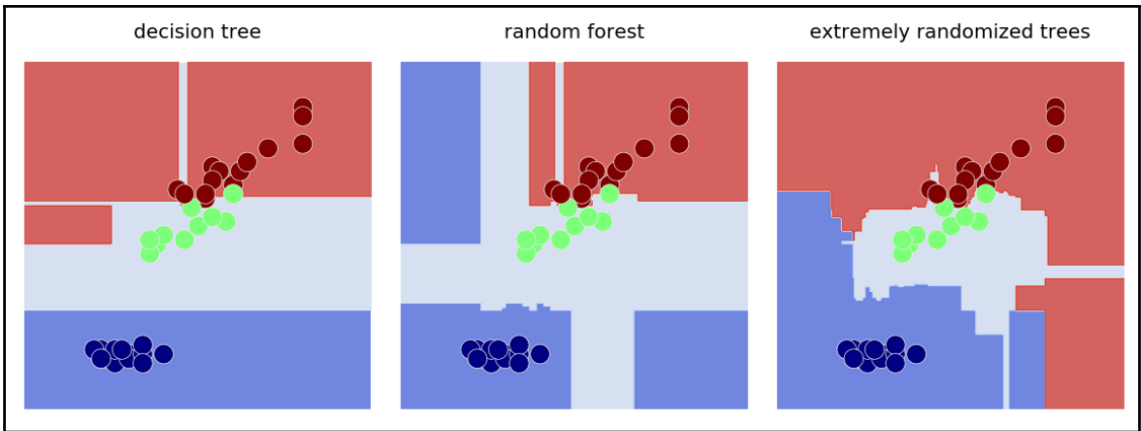
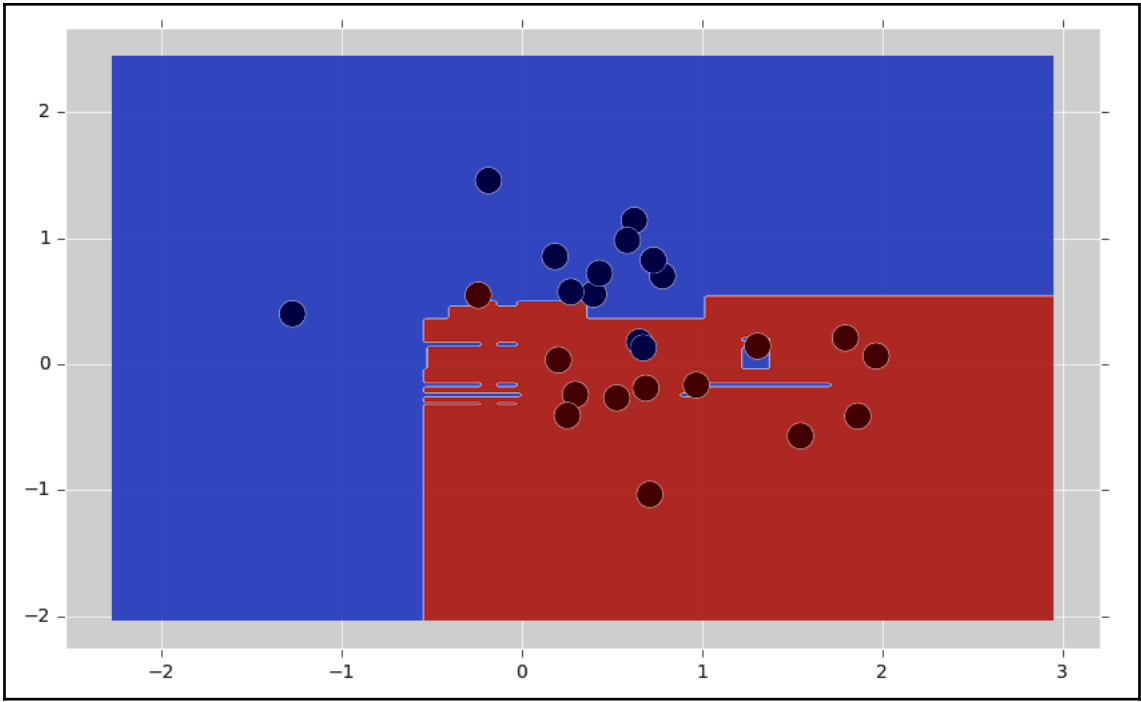
Chapter 10: Ensemble Methods for Classification

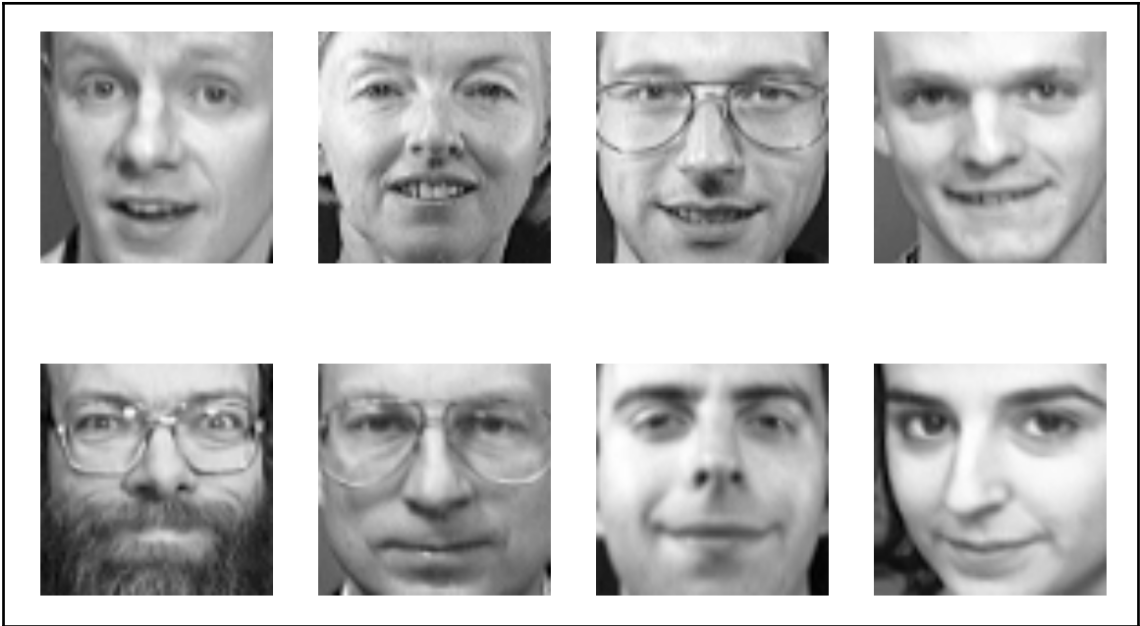


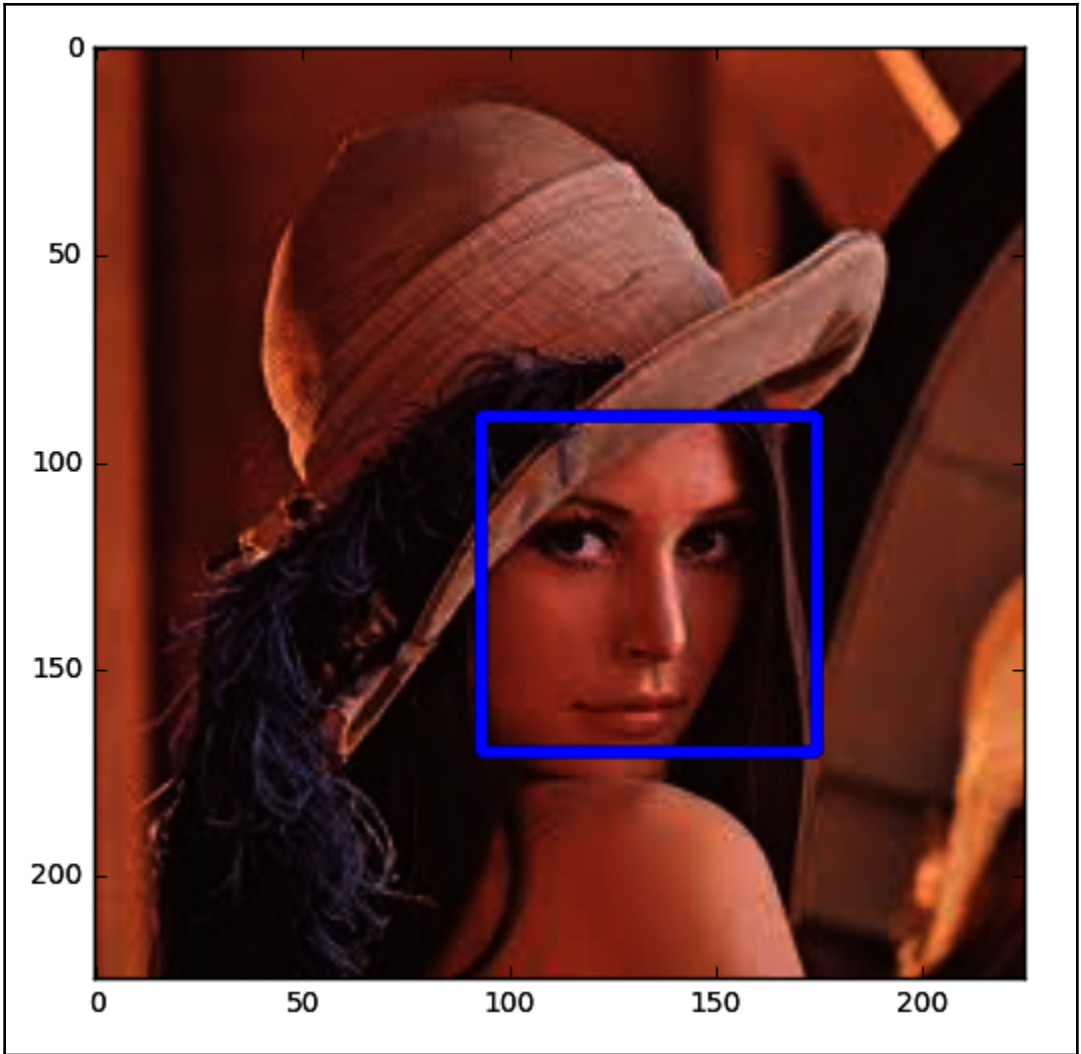




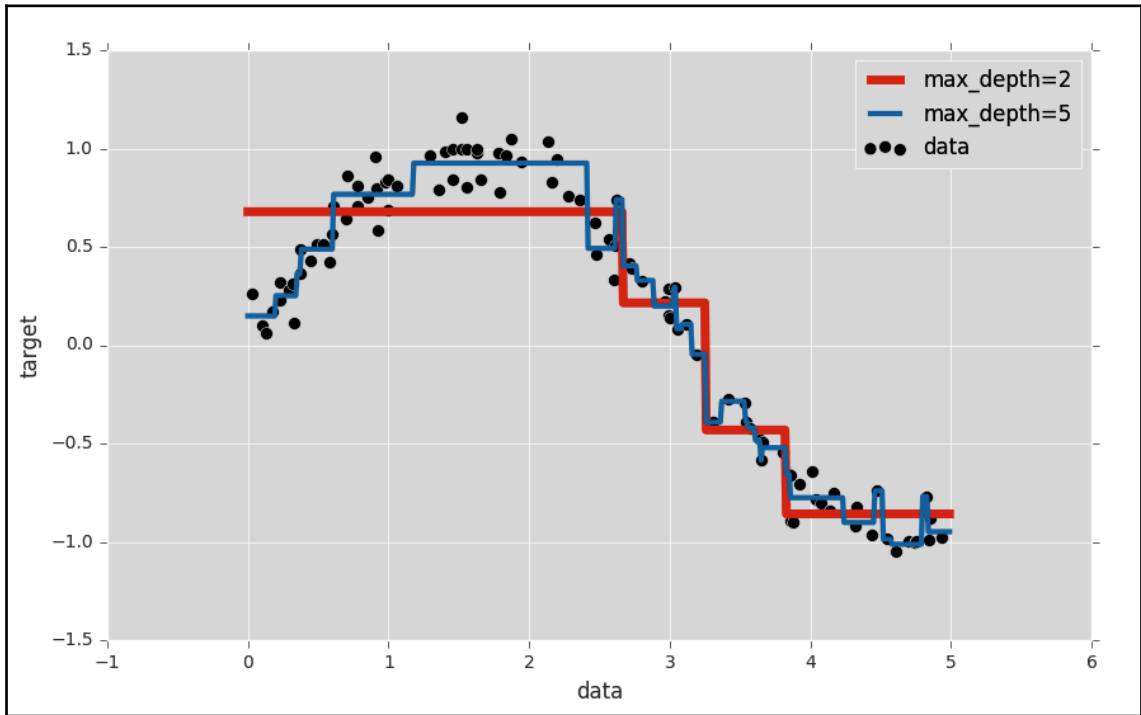


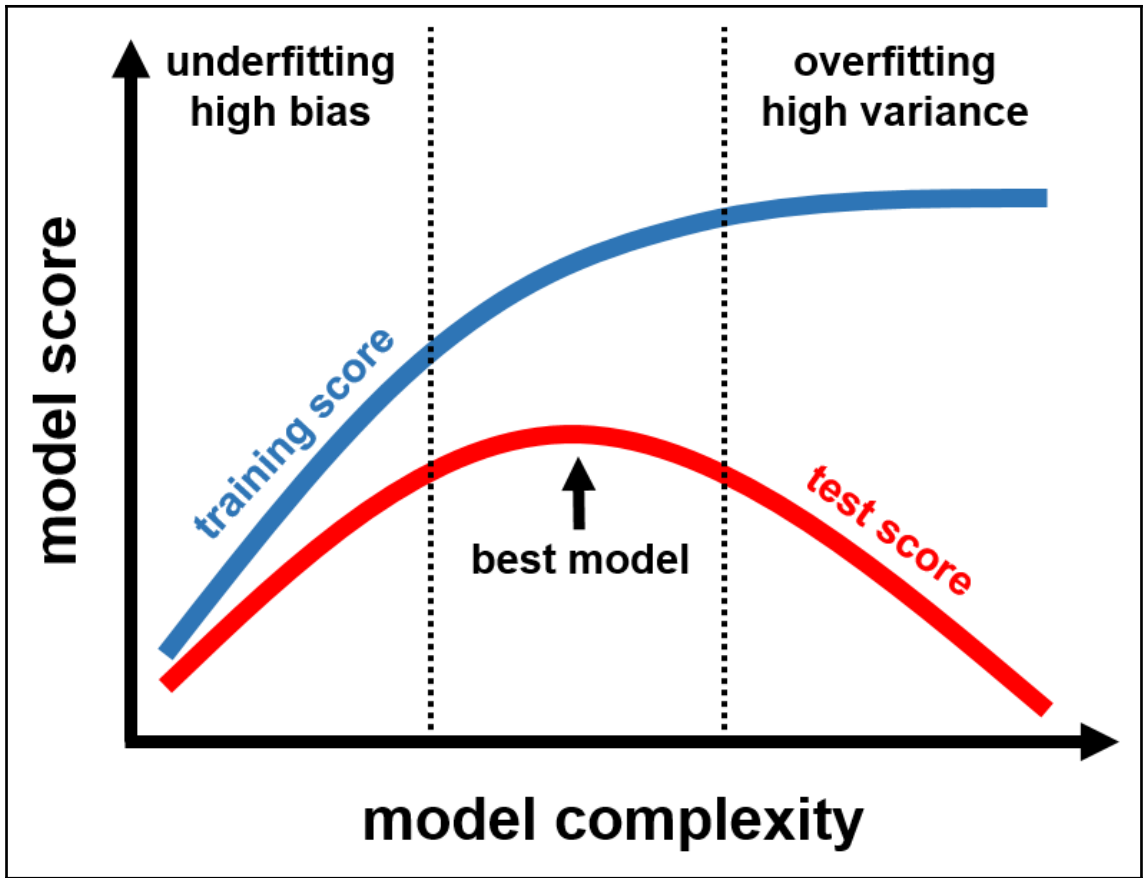


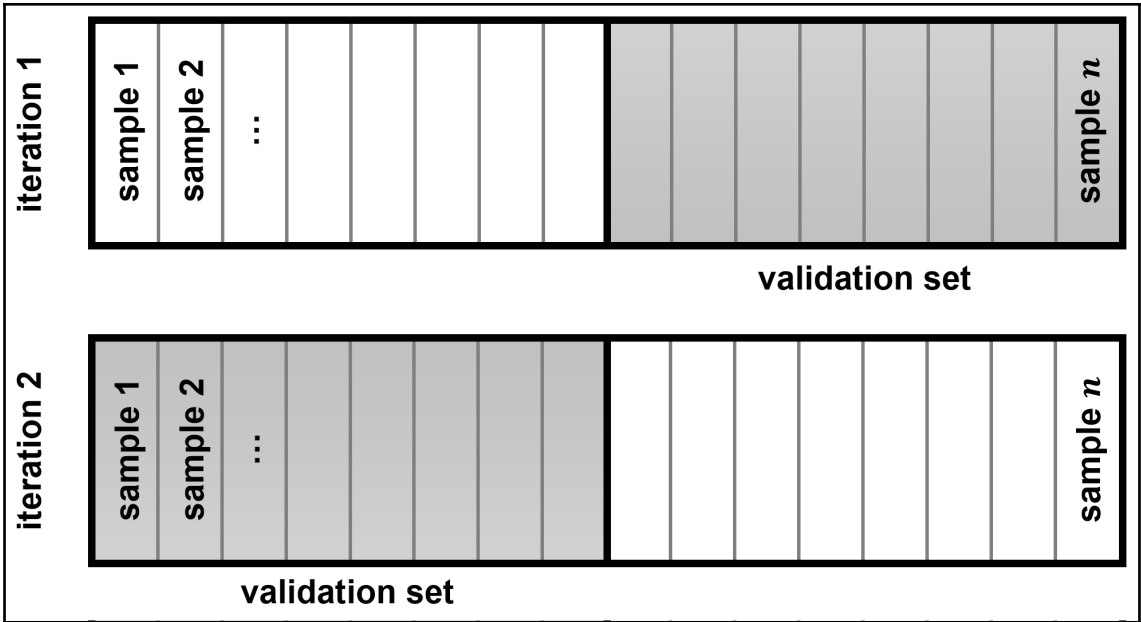


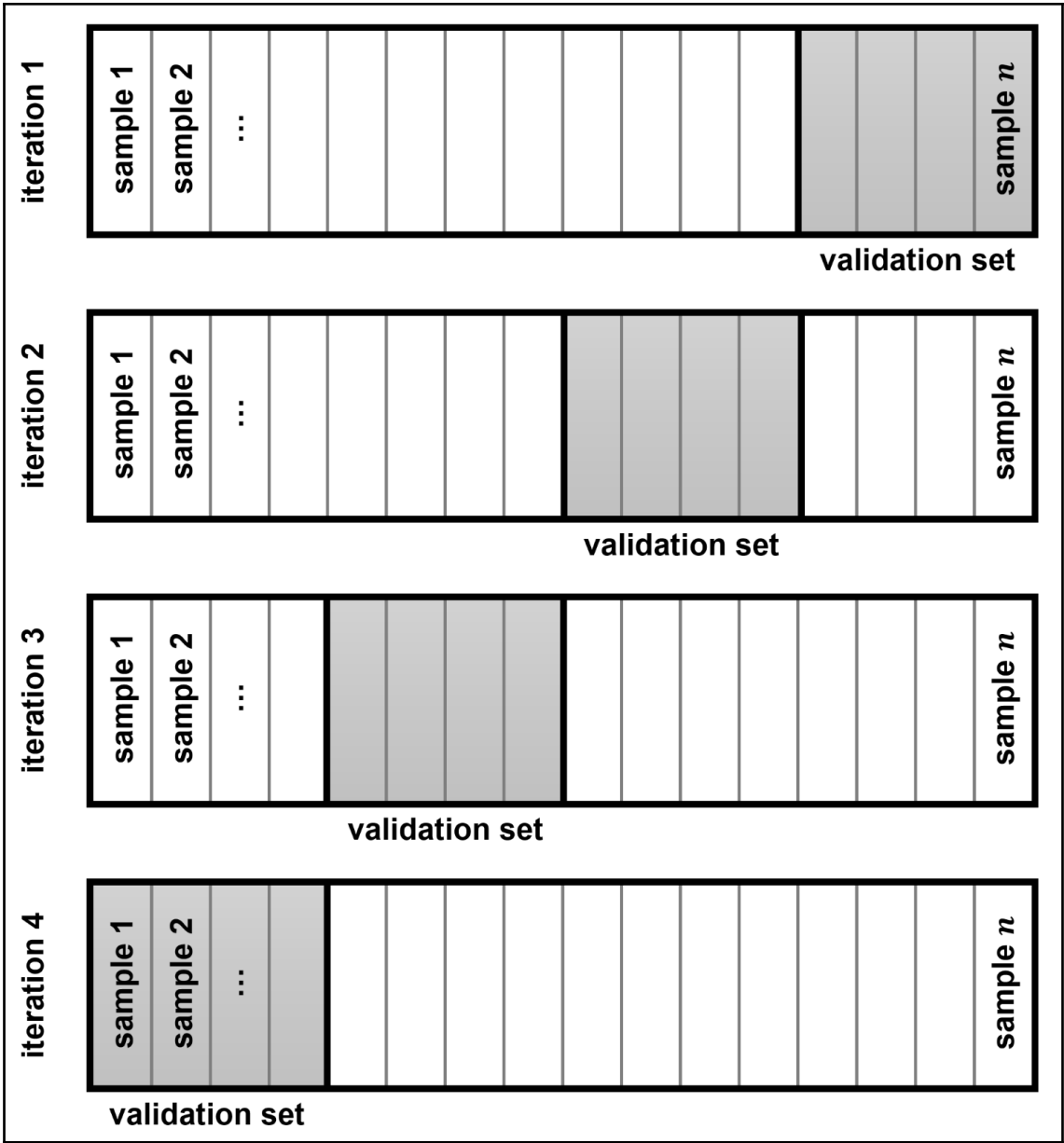


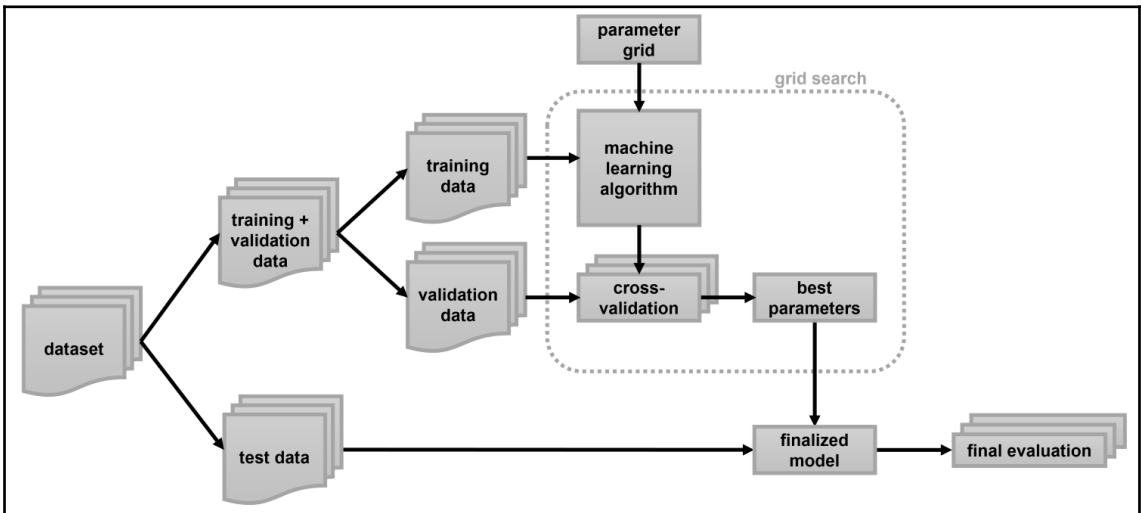
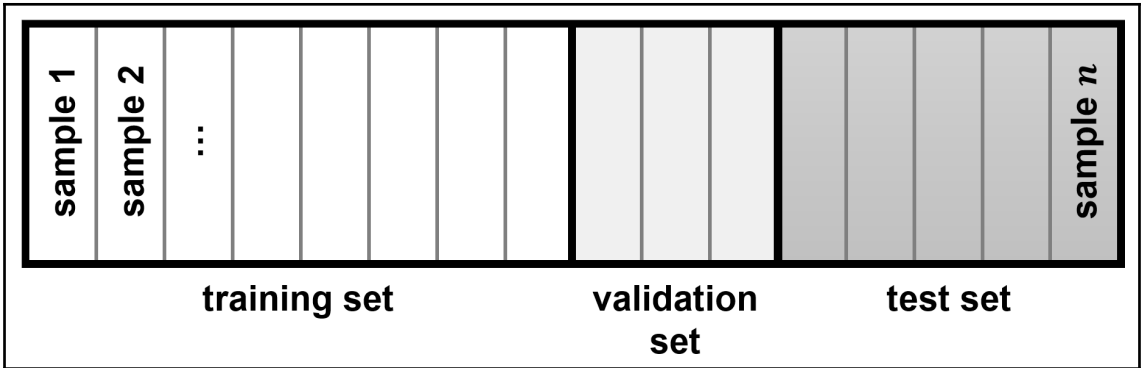
Chapter 11: Selecting the Right Model with Hyperparameter Tuning

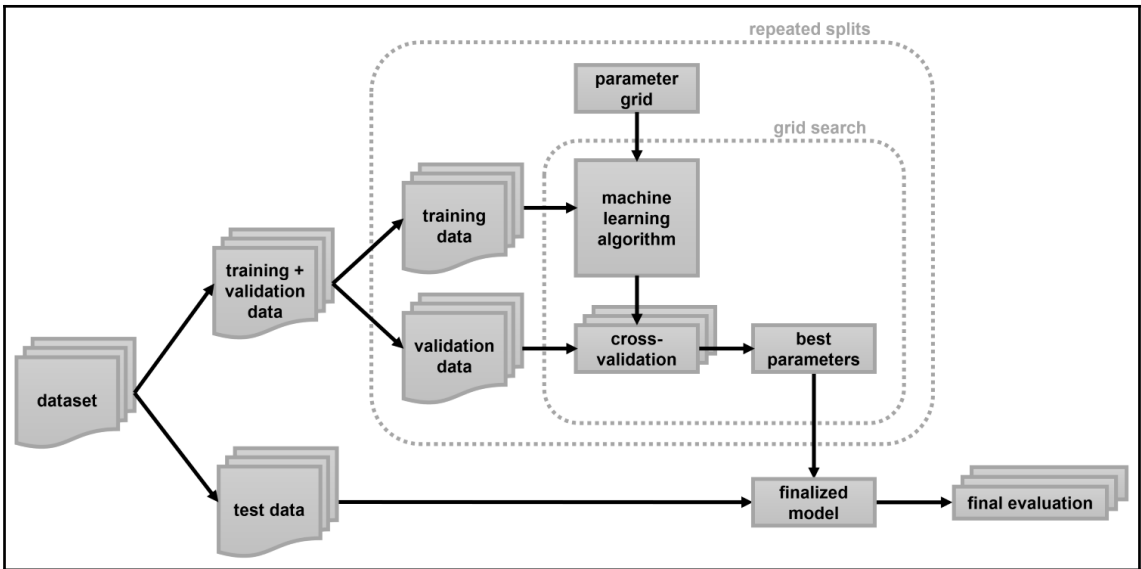




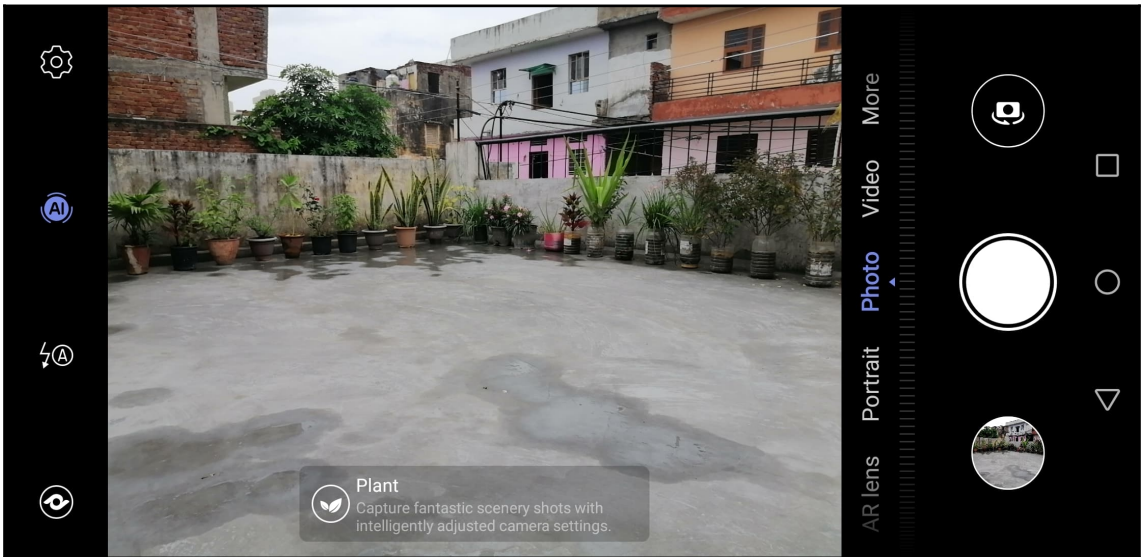








Chapter 12: Using OpenVINO with OpenCV



```
Top 10 results:
Image /opt/intel/computer_vision_sdk_fpga_2018.2.298/deployment_tools/demo/./demo/car.png
817 0.8363345 label sports car, sport car
511 0.0946488 label convertible
479 0.0419131 label car wheel
751 0.0091071 label racer, race car, racing car
436 0.0068161 label beach wagon, station wagon, wagon, estate car, beach waggon, station waggon, waggon
656 0.0037564 label minivan
586 0.0025741 label half track
717 0.0016069 label pickup, pickup truck
864 0.0012027 label tow truck, tow car, wrecker
581 0.0005882 label grille, radiator grille

[ INFO ] Execution successful

#####

Demo completed successfully.
```



