# Machine Learning with Python

By Oliver Theobald

## Chapter 1: Introduction

* Reminder: in ML the independent variables are denoted with a capital ‘X’ and the dependent variable is denoted with a lowercase ‘y’
* Self-learning can be divided into three categories
  + Supervised learning
    - Decodes known relationships between independent variables and the dependent variable
    - X\_train, y\_train, X\_test, and y\_test are all known/labeled
    - Classic example is usually predicting house cost given features about the house
  + Unsupervised learning
    - The dependent variable isn’t known/labeled but the model looks at patterns among independent variables to create a new output
    - i.e. dimensionality reduction
    - often used for detecting fraudulent activity
  + Reinforcement learning
    - Used for performing a sequence of decisions like chess
    - The opposite of unsupervised learning, the outputs are known but the inputs are unknown
    - The output can be the intended goal, i.e. win the chess game
    - Random input is fed and the behavior is graded
    - Overtime the model leverages feedback to achieve its desired output goal

## Chapter 2: Development Environment

* Steps on how to use Jupyter Notebook as the development environment with Python3

## Chapter 3: Machine Learning Libraries

* Pandas
  + ‘pandas’ comes from ‘panel data’
  + Managing and presenting panel data
* NumPy
  + ‘NumPy’ comes from ‘numeric Python’
  + Managing multi-dimensional arrays and matrices
  + Merging and slicing data sets
  + Mathematical functions
  + Said to perform faster than pandas with 50k rows or less
  + NumPy is better with numerical data frames whereas pandas can handle more mixed types
* Scikit-learn
  + Core library for machine learning
  + Algorithms and evaluation metrics
* Matplotlib
  + Visualization library that isn’t too advances
  + Often used with Seaborn library for better themes
* Seaborn
  + Visualization library
  + Preset themes for publication quality graphs
* TensorFlow
  + Google’s TensorFlow is the library of choice for ANN and deep learning
  + TensorFlow is only compatible with the Nvidia GPU which is no longer on Mac OS X
  + Mac OS X users can still use their CPU (slower) otherwise they need a cloud GPU

## Chapter 4: Exploratory Data Analysis

* Use pandas to explore dataframes
* EDA = exploratory data analysis
* Jupyter notebook exercise

## Chapter 5: Data Scrubbing

* Jupyter notebook exercise
* Steps
  + Remove unwanted variables
    - Vars not compatible with the data set
    - Do not influence the outcome
    - Correlated variables
  + One-hot encoding
    - Since this expands data horizontally, you can choose to drop the first reference
    - i.e. drop the sex\_male column and keep the sex\_female column, you already have all the necessary info
  + Drop missing values
    - Missing data can be split into three categories
      * Missing Completely at Random (MCAR)
        + No relationship between missing values and other values in a data set
      * Missing at Random (MAR)
        + Missing value is not related to its own value but to the value of other variables
        + i.e. in a census survery someone might skip a question because it was long and relevant info was already input somewhere else too
      * Nonignorable
        + Absence of data due directly to its own value or significance of the information
        + i.e. respondents with a criminal record may decline to supply information to certain questions due to feelings of sensitivity towards that question
* dimension reduction
  + transform data to a lower dimension, lessen computational resources and visualize patterns

## Chapter 6: Pre-model Algorithms

* As an extension of the data scrubbing process, unsupervised learning algorithms are sometimes used in advance of a supervised learning algorithm
* Principal component analysis (PCA)
  + Aka general factor analysis
  + Dramatically reduce data complexity and visualize data in fewer dimensions
  + Recreates dimensions as a linear combination of features called components and then ranks components that contribute most to patters in the data
  + Must standardize and scale the data!
* StandardScaler is often used with PCA, kNN, and SVM to rescale and standardize data features
  + Without, PCA is likely to lock onto features that maximize variance but could be exaggerated by another factor
* k-means clustering
  + identify groups of data points without prior knowledge of existing classes
  + splits dataset into k number of clusters
  + each cluster is assigned a random centroid
  + remaining data points are assigned to the closest centroid and the centroid coordinates are updated

## Chapter 7: Split Validation

* Data is usually split 30/70 or 20/80 for training/testing
* since the test data cannot be used to optimize the model, data scientists sometimes use a third independent data set called the validation set
* after building an initial model the validation dataset can be fed to model and used as feedback to optimize the model’s hyperparameters
* to maximize data, it’s possible to reuse the validation and test data as training data
  + bundle the used data with the original training data to optimize the model just before it’s put to use
  + However, since the validation and test set have been used for training, it can no longer be used as a validation or test set

## Chapter 8: Model Design

* High level overview at full procedure of building ML model
  + Import libraries
  + Import dataset
  + Exploratory data analysis
  + Data scrubbing
  + Pre-model algorithms
  + Split validation
  + Set algorithm
  + Predict
  + Evaluate
  + Optimize
* Evaluation methods
  + Accuracy score
    - Predicted correctly / total cases
    - Tends to hide lop-sided number of false-positives or false-negatives
  + Confusion matrix
    - Error matrix
    - Summarizes performance using false-positives and false-negatives
    - Accuracy = 1 – ((false-positive + false-negative) / total data points)
  + Classification report
    - Generates precision, recall, F1-score, support
    - Precision
      * # true-positives / # positively predicted cases
    - Recall
      * # true-positives / # of actual positives
  + F1-score
  + Support

## Chapter 9: Linear Regression

## Chapter 10: Logistic Regression

* Logistic regression accepts continuous and discrete variables as input variables to predict a qualitative outcome
* Best for predicting binary cases
  + Other techniques including the Naïve Bayes’ classified and SVM are considered to be more effective at classifying multiple discrete outcomes
* Analyzes relationships between variables and assigns probabilities to discrete outcomes using the Sigmoid function which converts numerical results into an expression of probability between 0 and 1
  + 0 = no chance
  + 1 = certain chance
* After assigning points to a class using the sigmoid function, a hyperplane is used as a decision boundary to split the two classes
  + The decision boundary can then be used to predict the class of future data points

## Chapter 11: Support Vector Machines (SVM)

* SVM is one of the best classifiers in ML for analyzing complex data and downplaying the influence of outliers
* Classify categorical outcomes
* Key feature is a margin, which is the distance between the boundary line and the nearest data point multiplied by 2
* The margin provides support to cope with new data points and outliers that would otherwise infringe on a logistic regression boundary line
* Hyperparameter C controls the cost of misclassification on the training data
  + Regulates the extent to which misclassified cases are ignored
  + Soft margin
  + Lower C = more errors permitted
* Hyperparameter gamma refers to the Gaussian radial basis function and the influence of the support vector
  + Small gamma = high bias, low variance
  + Large gamma = low bias, high variance

## Chapter 12: *k*-Nearest Neighbors

* Classifies unknown data points closest to the target point
* K is the number of nearest data points to look at
* Sometimes called a “memory-based procedure” because the full training data is used each time a prediction is made
  + Not good for large datasets and measuring multiple distances in high-dimensional data
* Reducing the number of dimensions, like in PCA, is a common way to simplify and prepare a dataset for k-NN

## Chapter 13: Decision Trees

* Cart (classification and regression trees)
* Tree methods: decision trees, bagging, random forests, boosting
  + All supervised learning
* Decision trees are prone to overfitting
* Random forests
  + Multiple decision trees using a randomized selection of input
  + Combine results by averaging/voting output
  + Less variance and overfitting than a decision tree
* Gradient boosting
  + Regression and classification technique
  + Sequential method
  + Aims to improve performance with each iteration
  + Evaluate performance of weak model and then overweighting subsequent models to reduce misclassifications
  + Learns from mistakes