Python Machine Learning Notes

1. Machine learning mainly has 3 categories: supervised, unsupervised and reinforcement learning;
2. Supervised learning refers to a set of samples where the desired output signals (labels) are already known;
3. Reinforcement learning is to develop an agent that improves its performance based on interactions with the environment;
4. Unsupervised learning is to explore the structure of our data to extract meaningful information without guidance of a known outcome variable or reward function;
5. Dimensionality reduction is used in unsupervised learning to remove noise from the data, which can also degrade the predictive performance of certain algorithms;
6. Sample is each instance in our epochs, and it has multi-dimensional features, the target output value is called label;
7. Machine learning programming should go through pre-processing stage: where we input and divide labels and raw data into training dataset & test dataset; then learning phrase where we apply algorithm to build the optimal model; the next one is evaluation, using reserved test dataset to test our final model to estimate the performance of our output models; Eventual stage is prediction, where we use our model to handle real world data and predict the output labels;
8. CSV stands for Commas Separate Values, we can use pandas' pd.read_csv('path', header = None) to import our pre-processing data;
9. Single layer neuron: multiple signals arrive at the dendrites, are then integrated into the cell body, and, if the accumulated signal exceeds a certain threashold, an output signal is generated and will be passed on by the axon;
10. For basic single layer neuron, unit step function is its decision function, in machine learning, the negative threshold is usually called the bias unit;
11. Basic perceptron learning rule: compute the output value net_output, calculate the error and update the weights; ADAptive LInear NEuron (Adaline) add the cost function, using the gradient descent to find the minimum cost that the cost function can converge. If the learning rate is wrong, it may overshoot the cost, we need to choose proper learning rate for Adaline learning algorithm;
12. Standardisation can optimise the sample's normal distribution, which help the gradient descent learning to converge more quickly. Simple just subtract the sample mean from every training sample and divide it by its standard deviation; (page 42)
13. When the dataset can go beyond millions, stochastic gradient descent can help to converge quicker to the minimum cost but it doesn't reach the global minimum, stochastic gradient descent can also used for online learning, when a model is trained on the fly as new training data arrives from time to time;
14. 5 steps to choose a classification algorithm: 1) selecting features and collecting training samples; 2) Choosing a performance metric; 3) choosing a classifier and optimization algorithm; 4) Evaluating the performance of the model 5) Tuning the algorithm
15. SciKit Learn has Iris dataset, after import datasets from sklearn, using datasets.load_iris() can load the data into a programme; from sklearn.model_selection import train_test_split can help us split datasets into training and test sets, they will also be automatically shuffled; from

sklearn.preprocessing import Standardscaler, we can standardize our datasets; from sklearn.linear_model import logisticRegression, we can using the built-in logisticRegression classifier directly;

16. Print string and array at one-line, in Python 2.7.15 version, we just need commas to separate them, for instance:
    demo = [1, 2, 3, 4]
    print 'Result is: ', demo [: 2]
    OUTPUT: [1,2,3];
17. Perceptron & Adaline has the drawback: they never converge unless the samples need to be linearly separable, otherwise the weight update won't stop unless we set up maximum epochs; Logistic Regression is a classifier, not regression. Its cost function is a S-shape sigmoid function, its logit function takes as input values in the range 0 to 1 and transforms them to values over the entire real-number range. If we predict correctly to 0 or 1, the cost on the y-axis approaches 0 or 1, However, if the prediction is wrong, the cost goes towards infinity (main point we penalize wrong predictions with an increasingly larger cost);
18. Overfitting means a model performs well on training data but does not generalize well to unseen data (test data), if a model suffers from overfitting, it has a high variance, it's basically "too sensitive about randomness"; Similarly, underfitting means a model is not complex enough to capture the pattern in the training data well and therefore also suffers from low performance on unseen data (high bias), we need to fine tune a model to get a "good compromise";
19. Regularisation is a very useful method to handle collinearity (high correlation among features), filter out noise from data and eventually prevent overfitting. The concept behind regularisation is to introduce additional information (bias) to penalize extreme parameter (weight) values. The term C parameter in sklearn's Logistic Regression classifier is directly related to the regularisation parameter, which is its inverse. By regularisation, the weight coefficients shrink if we decrease the parameter C, that is, if we increase the regularisation strength;
20. Super vector machines (SVMs): The margin is defined as the distance between the separating hyperplane (decision boundary) and the training samples that are closet to this hyperplane, which are the so-called support vectors, the objectives are maximize the margin;
21. For SVMs, the rationale behind having decision boundaries with large margins is that they tend to have a lower generalization error whereas models with small margins are more prone to overfitting; To deal with non-linearly separable case using SVMs with slack variables;
22. Logistic regression vs SVMs: In practice, they yield very similar results. Logistic regression tries to maximize the conditional likelihoods of the raining data, which makes it more prone to outliers than SVMs, which mostly care about the points that are closest to the decision boundary (SVMs). On the other hand, logistic regression has the advantage that it is a simpler model and can be implemented and updated more easily, which is attractive when working with streaming data;
23. The term kernel can be interpreted as a similarity function between a pair of samples. The minus sign inverts the distance measure into a similarity score, and due to the exponential term, the resulting similarity score will fall into a range between 1 (for exactly similar samples) and 0 (for very dissimilar samples);

24. Decision Tree's objectives are gaining the maximizing information gain, there are 3 commonly used impurity measures criteria: Gini impurity, entropy and classification error. The deeper the decision tree, the more complex the decision boundary becomes, which can easily result in overfitting; combining multiple decision trees can form the random forest classifier, it can be considered as an ensemble of decision trees. Individual decision Tree suffers from high variance, to build a more robust model that has a better generalization performance and is less susceptible to overfitting;
25. Random forest without replacement: 2, 1, 3, 4, 0; random forest with replacement: 1, 3, 3, 4, 1. It basically depends whether the original samples will be reused or not;
26. Parametric vs non-parametric models: ML algorithms can be classified as parametric & non-parametric models. Using parametric models, estimate parameters from the training dataset to learn a function that can classify new data points without requiring the original training dataset anymore. Typical parametric models: perceptron, logistic regression and the linear SVM. In contrast, non-parametric models can't be characterized by a fixed set of parameters, and the number of parameters grows with the raining data. Typical non-parametric models: decision tree, random forest, kernel SVM and KNN (instance-based learning).
27. For missing data, we can use dropna function from pandas library to remove those rows containing null data. In practice, it's considered good practice to provide class labels as integer arrays to avoid technical glitches. If we can imputing them using *mean, median or most_frequent* mathematic methods. For nominal (has no orders) & ordinal (has orders), we can use encoding to mapping ordinal/nominal features into integers, however, one issue will emerge after mapping: the computer will deem the encoded features have less or greater relationships (e.g. green => 0, red => 2, after mapping, the computer thinks the red > green). A method called one-hot encoding, it creates a new dummy features for each unique value in the nominal feature volume.
28. L2 regularization (square of the weights) is one approach to reduce the complexity of a model by penalizing large individual weights, whereas L1 regularization (Sum of absolute values of the weights) can help us to avoid overfitting by reducing the complexity of a model. The goals of them are to find the combination of weight coefficients that minimize the cost function for the training data with minimum penalty. view book page 124 - 125.
29. Sequential feature selection algorithms: an alternative way to reduce the complexity of the model and avoid overfitting is dimensionality reduction via feature selection, which is useful for unregularized models. Two main categories of dimensionality reduction techniques: feature selection and feature extraction. Sequential Backward Selection (SBS) aims to reduce the dimensionality of the initial feature subspace with a minimum decay in performance of the classifier to improve upon computational efficiency. Random forest can be used to rank features by their respective importance measures.
30. A positive covariance between two features indicates that the features increase or decrease together, whereas a negative covariance indicates that the features vary in opposite directions. covariance is used to tell us the sparsity of two features intuitively.

31. There are 3 fundamental dimensionality reduction techniques for feature extraction: standard principle component analysis (SPCA), Linear Discriminants Analysis (LDA) and Kernel PCA.
32. Using PCA, we projected data onto a lower-dimensional subspace to maximize the variance along the orthogonal feature axes, while ignoring the class labels.
33. LDA, in contrast to PCA, is a technique for supervised dimensionality reduction, which means that it considers class information in the training dataset to attempt to maximize the class-separability in a linear feature space.
34. Kernel PCA, using the kernel trick and a temporary projection into a higher-dimensional feature space, you were ultimately able to compress datasets consisting of nonlinear features onto a lower-dimensional subspace where the classes became linearly separable.