Task2- References – code

<结合Scikit-learn介绍几种常用的特征选择方法>

<http://www.17bigdata.com/%E7%BB%93%E5%90%88scikit-learn%E4%BB%8B%E7%BB%8D%E5%87%A0%E7%A7%8D%E5%B8%B8%E7%94%A8%E7%9A%84%E7%89%B9%E5%BE%81%E9%80%89%E6%8B%A9%E6%96%B9%E6%B3%95.html>

<专家说，这40道面试题都答上来，你就能去机器学习初创公司当数据科学家了>

<http://jiasuhui.com/archives/123488>

<drop column>

<http://erikrood.com/Python_References/dropping_rows_cols_pandas.html>

<数据挖掘算法学习（八）Adaboost算法>

<https://wizardforcel.gitbooks.io/dm-algo-top10/content/adaboost.html>

<confidence interval….. regression>

<http://markthegraph.blogspot.com.au/2015/05/using-python-statsmodels-for-ols-linear.html>

< Evaluating a Classification Model/>

<http://www.ritchieng.com/machine-learning-evaluate-classification-model/>

<NB>

<http://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html#sklearn.naive_ba>yes.GaussianNB

<NN>

<http://scikit-learn.org/stable/modules/neural_networks_supervised.html>

<【机器学习】Random Forest(随机森林)入门和实战（三）什么时候使用>

<http://blog.csdn.net/Lin_Ting/article/details/55097766>

<http://blog.csdn.net/ac540101928/article/details/51689505>

<Bagging（Bootstrap aggregating）、随机森林（random forests）、AdaBoost>

<http://blog.csdn.net/xlinsist/article/details/51475345?locationNum=1&fps=1>

<http://www.csdn.net/>

<http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html#sklearn.neural_network.MLPClassifier>

<color>

<https://xkcd.com/color/rgb/>

**<Bagging is better performed in imbalanced data>**

[**http://ieeexplore.ieee.org/document/5645694/?reload=true&arnumber=5645694**](http://ieeexplore.ieee.org/document/5645694/?reload=true&arnumber=5645694)

The ROC curve is insensitive to changes in class imbalance; see Fawcett (2004) "ROC Graphs: Notes and Practical Considerations for Researchers".

<Tackling the Poor Assumptions of Naive Bayes Text Classifiers>

http://people.csail.mit.edu/jrennie/papers/icml03-nb.pdf

<Thoughts on Machine Learning – Dealing with Skewed Classes>

<https://florianhartl.com/thoughts-on-machine-learning-dealing-with-skewed-classes.html>

<>

<https://www.kaggle.com/bmanohar16/glass-ensemble-classification-a-case-study>

**<Essentials of Machine Learning Algorithms (with Python and R Codes)>**

<https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>

<https://www.analyticsvidhya.com/blog/2017/02/top-28-cheat-sheets-for-machine-learning-data-science-probability-sql-big-data/>

**<Tree>**

<https://www.analyticsvidhya.com/blog/2016/04/complete-tutorial-tree-based-modeling-scratch-in-python/>

<https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-tree-based-models/>

<PCA>

<https://www.analyticsvidhya.com/blog/2016/03/practical-guide-principal-component-analysis-python/>

<Dimension reducion>

<https://www.analyticsvidhya.com/blog/2015/07/dimension-reduction-methods/>

<Clusting>

<https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/>

**<Books>**

<https://www.analyticsvidhya.com/blog/2016/10/18-new-must-read-books-for-data-scientists-on-r-and-python/>

**<Project samples>**

<https://www.analyticsvidhya.com/blog/2016/10/17-ultimate-data-science-projects-to-boost-your-knowledge-and-skills/>

<classification - Loan Prediction Data Set>

<https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-learn-data-science-python-scratch-2/>

<regression - Bigmart Sales Data Set>

<https://www.analyticsvidhya.com/blog/2016/02/bigmart-sales-solution-top-20/>

<data manipulation>

<https://www.analyticsvidhya.com/blog/2016/01/12-pandas-techniques-python-data-manipulation/>

<EDA & ML beginner>

<https://www.analyticsvidhya.com/blog/2016/09/18-free-exploratory-data-analysis-tools-for-people-who-dont-code-so-well/>

<https://www.analyticsvidhya.com/blog/2016/10/16-new-must-watch-tutorials-courses-on-machine-learning/>

**RESCALING** attribute data to values to scale the range in [0, 1] or [−1, 1] is useful for the optimization algorithms, such as gradient descent, that are used within machine learning algorithms that weight inputs (e.g. regression and neural networks). Rescaling is also used for algorithms that use distance measurements for example K-Nearest-Neighbors (KNN). Rescaling like this is sometimes called "normalization". MinMaxScaler class in python skikit-learn does this.

**NORMALIZING** attribute data is used to rescale components of a feature vector to have the complete vector length of 1. This is "scaling by unit length". This usually means dividing each component of the feature vector by the Euclidiean length of the vector but can also be Manhattan or other distance measurements. This pre-processing rescaling method is useful for sparse attribute features and algorithms using distance to learn such as KNN. Python scikit-learn Normalizer class can be used for this.

**STANDARDIZING** attribute data is also a preprocessing method but it assumes a Gaussian distribution of input features. It "standardizes" to a mean of 0 and a standard deviation of 1. This works better with linear regression, logistic regression and linear discriminate analysis. Python StandardScaler class in scikit-learn works for this.

<preprocessing>

<https://www.analyticsvidhya.com/blog/2016/07/practical-guide-data-preprocessing-python-scikit-learn/>

<https://www.analyticsvidhya.com/blog/2016/03/complete-solution-top-11-telstra-network-disruptions-kaggle-competition/>

<sentiment analysis>

<https://www.analyticsvidhya.com/blog/2016/02/step-step-guide-building-sentiment-analysis-model-graphlab/>

<improve accuracy>

<https://www.analyticsvidhya.com/blog/2015/12/improve-machine-learning-results/>

<boosting>

<https://www.analyticsvidhya.com/blog/2015/11/quick-introduction-boosting-algorithms-machine-learning/>

<ensemble>

<https://www.analyticsvidhya.com/blog/2015/09/questions-ensemble-modeling/>

<Github trick>

<https://www.analyticsvidhya.com/blog/2015/07/github-special-data-scientists-to-follow-best-tutorials/>

<Youtuber>

<https://www.analyticsvidhya.com/blog/2015/07/top-youtube-videos-machine-learning-neural-network-deep-learning/>

<kaggle>

<https://www.analyticsvidhya.com/blog/2015/06/start-journey-kaggle/>

<adaboost>

<https://www.analyticsvidhya.com/blog/2015/05/boosting-algorithms-simplified/>

<python tutorial- entry level>

<https://www.analyticsvidhya.com/blog/2015/04/pycon-montreal-2015-data-science-workshops/>

<model evaluation>

<https://www.analyticsvidhya.com/blog/2016/02/7-important-model-evaluation-error-metrics/>

<knn>

<https://www.analyticsvidhya.com/blog/2014/10/introduction-k-neighbours-algorithm-clustering/>

<SVM>

https://www.analyticsvidhya.com/blog/2014/10/support-vector-machine-simplified/