**Week 07: R to Python**

|  |  |  |  |
| --- | --- | --- | --- |
| **R** | **Python** | **Use** | **Difference** |
| complete.cases() | .dropna() | Apply to dataframe to drop any rows that have any NA’s | .dropna() has further arguments to specify what rows to drop based on what is desired  [Python](https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.dropna.html)  [R](https://www.statology.org/complete-cases-in-r/) |
| knn() | KNeighborsClassifier() | Implements k-nearest neighbor test, | knn() is from the library class and is more of an all in one function, while KneighborsClassifier has several other sklearn function to help perform knn tests.  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html)  [R](https://www.rdocumentation.org/packages/class/versions/7.3-20/topics/knn) |
|  | .fit() | Used in this case to fit the knn model to the specific training data | As mentioned above, this is one of the subsequent functions in the knn test in sklearn  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html) |
|  | .predict\_proba() | Predict the class probabilities in knn test | Same as above  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html) |
|  | .predict() | Predict the actual class in the knn test | Same as above  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html) |
| Ifelse() | numpy.where() | Used to return elements of one of two choices based on condition given | np.where() requires numpy to be imported.  [Python](https://numpy.org/doc/stable/reference/generated/numpy.where.html)  [R](https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/ifelse) |
| confusionMatrix() | metrics.confusion\_matrix | Creates a confusion matrix which can be used to evaluate the accuracy of a classification | Python function requires metrics to be imported. R function gives much more information, which can be obtained in python using more functions  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html)  [R](https://www.rdocumentation.org/packages/caret/versions/3.45/topics/confusionMatrix) |
|  | metrics.ConfusionMatrixDisplay.from\_predictions | Helps display the confusion matrix in a user-friendly manner | Metrics import from sklearn required  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.ConfusionMatrixDisplay.html) |
| prediction()  performance() | metrics.RocCurveDisplay.from\_predictions | Pass your probabilities and target to function. Outputs an ROC curve with the AUC value | In python, Metrics import from sklearn is required. In R, ROCR library necessary.  Also in R, performance() function is used to extract fpr, tpr and auc from the prediction object  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.RocCurveDisplay.html#sklearn.metrics.RocCurveDisplay.from_predictions)  [R](https://www.rdocumentation.org/packages/ROCR/versions/1.0-11/topics/prediction) (prediction)  [R](https://www.rdocumentation.org/packages/ROCR/versions/1.0-1/topics/performance) (performance) |
|  | metrics.roc\_auc\_score | Return auc value of ROC curve based on target values and probability of the positive class | Metrics import from sklearn required  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html) |
| LogLoss | metrics.log\_loss | Determine the log loss of prediction model | In Python, Metrics import from sklearn required. In R, MLmetrics library required  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.log_loss.html)  [R](https://www.rdocumentation.org/packages/MLmetrics/versions/1.1.1/topics/LogLoss) |
| F1\_Score() | metrics.f1\_score | Compute the F1 score, to help judge the performance of a classification model | In Python, Metrics import from sklearn required. In R, MLmetrics library required  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html)  [R](https://www.rdocumentation.org/packages/MLmetrics/versions/1.1.1/topics/F1_Score) |
| confusionMatrix() | metrics.classification\_report | Display several other metric factors like precision and recall | Need Metrics import.  Gives similar extra information that the confusionMatrix() function in R gives  [Python](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html)  [R](https://www.rdocumentation.org/packages/caret/versions/3.45/topics/confusionMatrix) |