**04: ML Concepts and Foundations**

**Functions**

|  |  |  |  |
| --- | --- | --- | --- |
| **R** | **Python** | **Purpose** | **Differences** |
| rbind() | sklearn.model\_selection.train\_test\_split() | Splitting a dataset into training and testing data | Python function is less complex in parameter inputs and permits easy control of the proportion of data that goes into the train or test datasets. |
| knn() | sklearn.neighbors.KNeighborsClassifier() | Training a knn model on test data (dependent and target variables) | They are very similar; the main differences lie in the functions that can be called/applied to the resulting model (in R, we checked the model summary and compared it to the target while we called model.score(independent, target)) |
| tapply() | df.groupby() | Finding the summary statistics for variables, grouped by species | These take different approaches for the same effect. In R, we group and apply the `summary` function to the designated column. In Python, we group the values in advance and call the functions in other lines. |

**Changes: Class Notes**

|  |  |
| --- | --- |
| **Difference** | **Change** |
| Libraries | We use different libraries in different languages; R’s are built-in, but we use sklearn in Python. |