Data Analytics Lab 3 Figures + Outputs

Classifier

** Used a preprocessor to make all weights valued equally and 10 fold cross validation to evaluate best model/k

Contingency Table and Accuracy of Weight based classifier
(inputs = whole_weight + shucked_wieght + viscera_wieght + shell_weight)

```
> # Get accuracy for the weight based classifier
> train.cm = as.matrix(table(Actual = knn.train.true, Predicted = knn.train.predicted))
> train.cm
       Predicted
Actual  young adult old
  young    20    12   0
  adult     7    33   5
  old       1     5  22
> train.accuracy <- sum(diag(train.cm))/nrow(train)
> train.accuracy
[1] 0.7142857
> test.cm = as.matrix(table(Actual = knn.test.true, Predicted = knn.test.predicted))
> test.cm
       Predicted
Actual  young adult old
  young   702   576  97
  adult   157   676 932
  old      36   268 627
> test.accuracy <- sum(diag(test.cm))/nrow(test)
> test.accuracy
[1] 0.492508
```

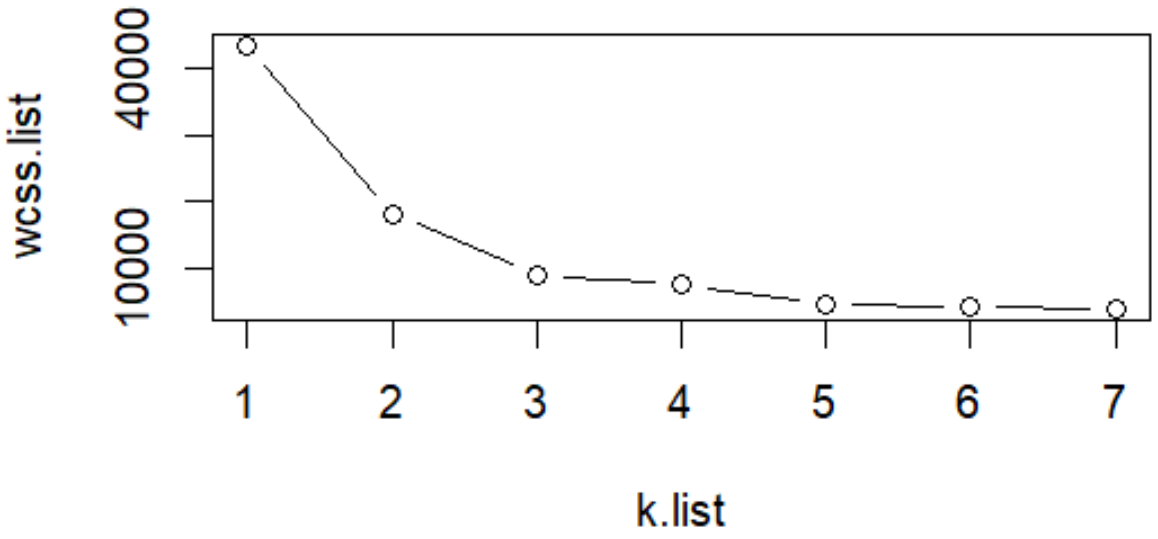
Contingency Table and Accuracy of Measurement based classifier
(inputs = length + diameter + height + rings)

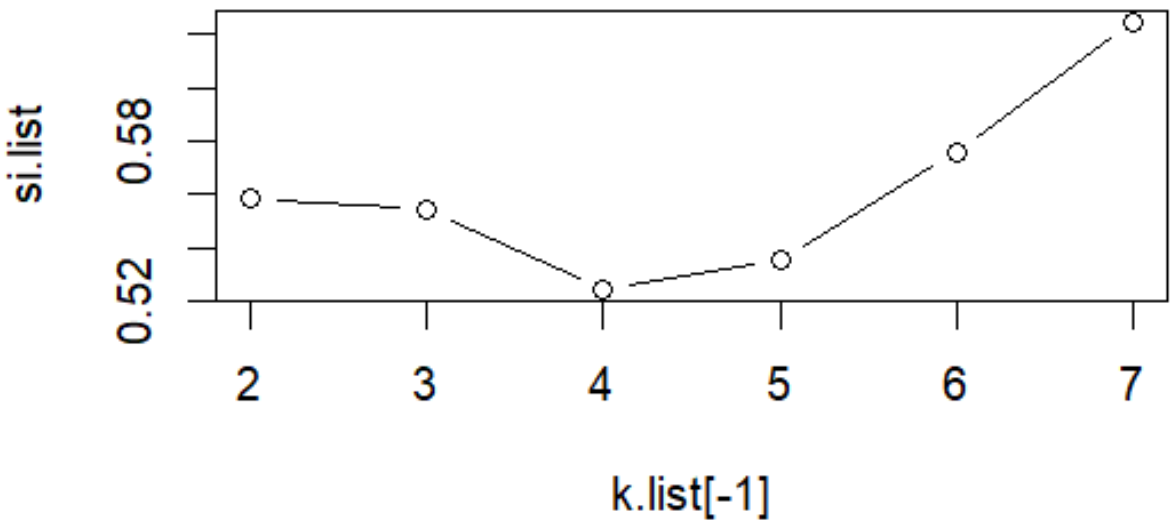
```
> # Get accuracy for measurement based classifier
> train.cm = as.matrix(table(Actual = knn.train.true, Predicted = knn.train.predicted))
> train.cm
       Predicted
Actual  young adult old
  young    28     4   0
  adult     1    43   1
  old       0     1  27
> train.accuracy <- sum(diag(train.cm))/nrow(train)
> train.accuracy
[1] 0.9333333
> test.cm = as.matrix(table(Actual = knn.test.true, Predicted = knn.test.predicted))
> test.cm
       Predicted
Actual  young adult  old
  young  1027   347    1
  adult    10  1571  184
  old       0   149  782
```

Train across k (done using grid) to evaluate optimal k. **Optimal k is 3**

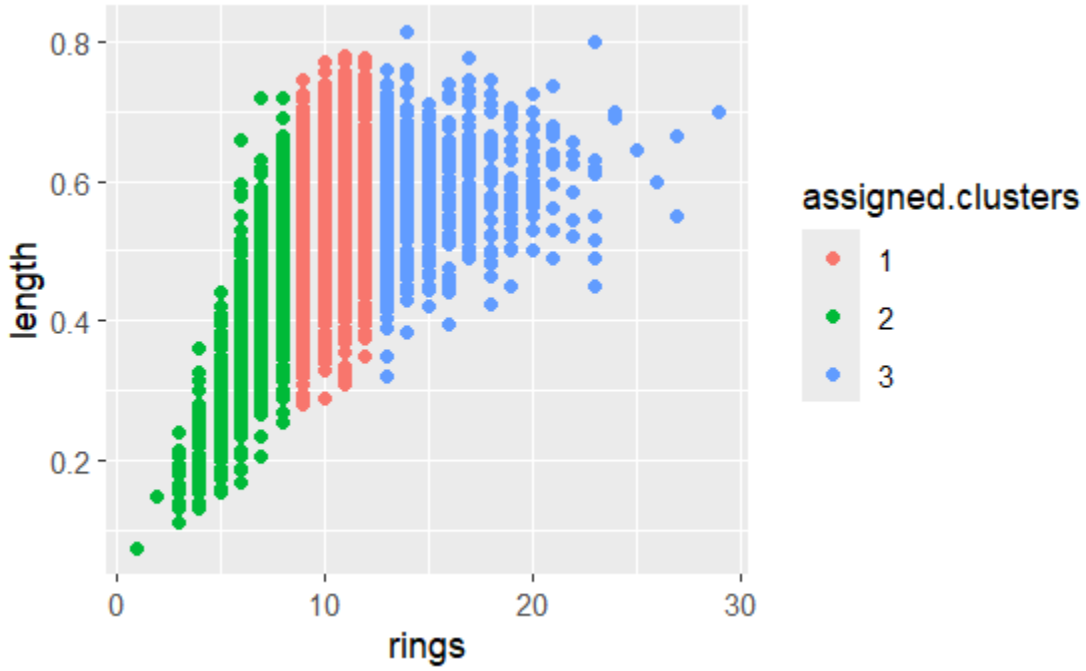
```
> grid <- expand.grid(k = c(3, 5, 7, 9, 11))
> # Train with the list of k values and pick the best based on the accuracy
> # Use preprocess so all values weighted equally and scaled
> mod.knn <- train(
+   age.group ~ length + diameter + height + rings,
+   data = train,
+   method = "knn",
+   preProcess = c("center", "scale"),
+   tuneGrid = grid,
+   metric="Accuracy",
+   trControl=ctrl
+ )
> # Best k for model
> mod.knn$bestTune
  k
1 3
> # Get the accuracy
> knn.train.predicted <- predict(mod.knn,train[,-10])
> knn.test.predicted <- predict(mod.knn,test[,-10])
> knn.train.predicted <- predict(mod.knn.meas,train[,-10])
> knn.test.predicted <- predict(mod.knn.meas,test[,-10])
> train.cm = as.matrix(table(Actual = knn.train.true, Predicted = knn.train.predicted))
> train.cm
       Predicted
Actual  young adult old
  young    28     4   0
  adult     1    43   1
  old       0     1  27
> train.accuracy <- sum(diag(train.cm))/nrow(train)
> train.accuracy
[1] 0.9333333
> test.cm = as.matrix(table(Actual = knn.test.true, Predicted = knn.test.predicted))
> test.cm
       Predicted
Actual  young adult  old
  young  1027   347    1
  adult    10  1571  184
  old       0   149  782
> test.accuracy <- sum(diag(test.cm))/nrow(test)
```

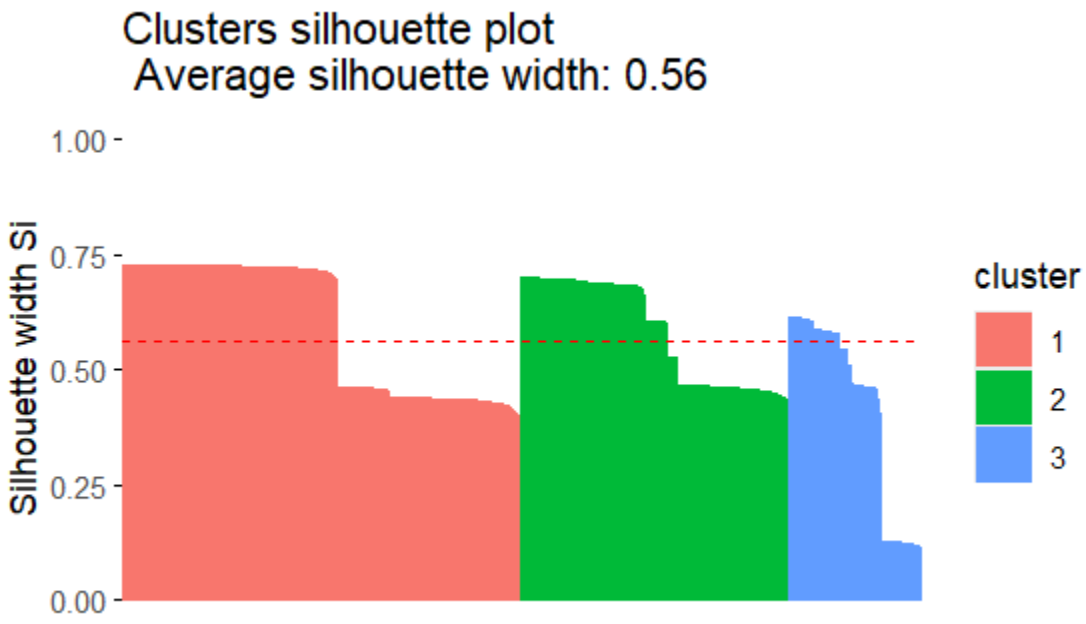
Clustering
Finding optimal k for kMeans



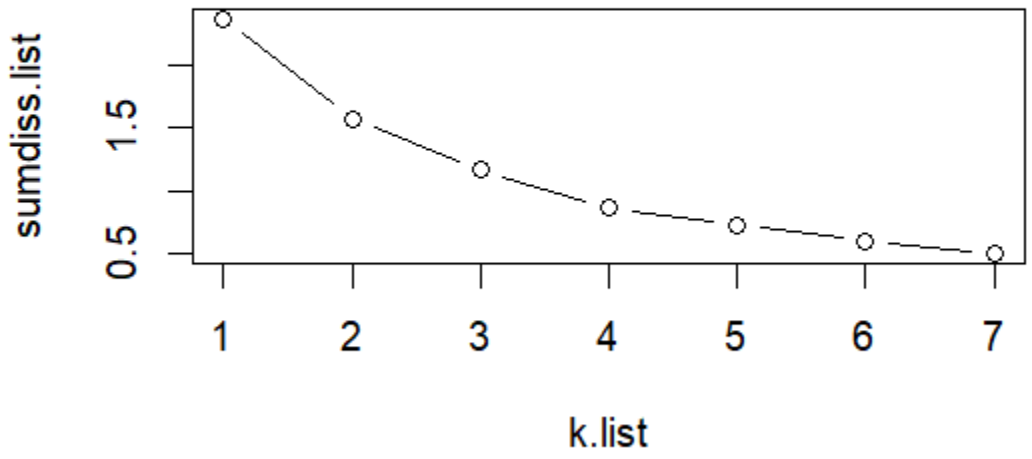


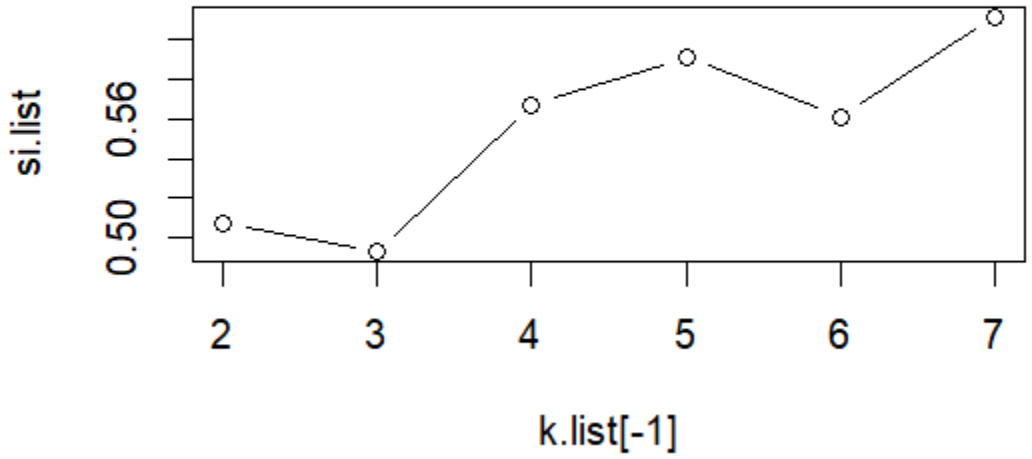
Clustering and Silhouette for optimal k = 3





Finding optimal k for PAM





Clustering and Silhouette for optimal k = 5