R is a programming language and software environment for statistical computing and graphics.

- Note: Many features of R now being incorporated into Python through packages like Pandas, SciPy, scikit, and others
- Main data type is "data frame", similar to relational table

Online demo using RStudio:

**Import Data**

```
Import Dataset generates commands:
TC <- read.csv("~/TempsCat.csv"); View(TC)
print(TC)
TC
summary(TC)
nrow(TC)
```

**For Loops, Data Frame Structure**

```
for (i in 1:nrow(TC)) { print(TC[i,]) }
for (i in 1:ncol(TC)) { print(TC[,i]) }
```

**Basic Data Operations**

- select columns
  
  `TC[, c('city','temp')]

- select rows
  
  `TC[TC$temp<20, ]

- select both
  
  `TC[TC$temp<20, c('city','temp')]

- select value by absolute row and column
  
  `TC[12,5]

- sort
  
  `TC[order(TC$1ng), ]

- aggregation
  
  `mean(TC$temp)
  `aggregate(TC$temp, by=list(TC$region), FUN=mean)

Add T$coastal to list

To avoid “TC$”

```
attach(TC)
```
Regression

cor(lat, temp)
reg <- lm(temp ~ lat)
plot(lat, temp)
abline(reg)

Scatterplot with Colored Categories, Gridlines, Temperature Labels

plot(lng, lat, xlab=rev(range(lng)),
    ylab=rev(range(lat)),
    col=c("green","orange","yellow","blue","red")[category], pch=16, cex=2)
grid(25,25)
text(lng, lat, labels=temperature, pos=1, offset=0.75)

Categorization (kNN)

Predict category from features region, coastal, lat, lng

Packages: class checkbox, or:
install.packages("class")
library(class)
TCCat <- transform(TC, region=as.numeric(region), coastal=as.numeric(coastal))
train <- TCCat[1:45, 3:6]
test <- TCCat[46:56, 3:6]
trainlabels <- TCCat[1:45, 8]
pred <- knn(train, test, trainlabels, k=7)
print(pred, max.levels=0)
print(TCCat[46:56, 8], max.levels=0)

Print side-by-side comparison and compute percentage correct

numtests <- nrow(test)
umcorrect <- 0
for (i in 1:numtests) {
  message("predicted:", pred[i], " actual:", TCCat[45+i,8]);
  if (pred[i] == TCCat[45+i,8]) numcorrect <- numcorrect+1
}
message("Fraction correct: ", numcorrect/numtests)

Also try with larger and smaller values of k, fewer features

Alternative for fraction correct: mean(pred == TCCat[46:56,8])

Clustering (k-Means)

crus <- kmeans(TC[, c('lng', 'lat')], 5)
TCClus <- data.frame(TC, crus$cluster)
with(TCClus, plot(lng, lat, xlab=rev(range(lng)), col=clus$cluster,
    pch=16, cex=2))
points(clus$centers, pch=8, cex=2)

Also try with larger number of clusters, change to cluster by 'temp'