Supervised Learning in R

Swagata Duari

Focus

- Linear Regression
- K-nearest Neighbor (knn)
- Decision Trees
- Naive Bayes
- Neural Network

Requirements

- R, RStudio
- **`Caret`** Package in R

- Install package using command

```
install.packages("caret", dependencies = true)
```

- Load package using command

```
library(caret)
```

Supervised Learning

It is the task of inferring a function from *labeled training data*

- Training a model
  - Instance-based learning, Naive Bayes, decision tree learning, etc.
- Testing a model
  - k-fold cross-validation
- Prediction using that model

Training a Model

Training

The `train` function is used to train a model

```
modelFit <- train(Class ~ ., data = Training,
        method = "xxx",
        trControl = fitControl)
```

- `Class` is the class labels
- `Training` is the training data
- `xxx`, specifies the type of model
- `fitControl` specifies the list of control parameters

List of Control Parameters

```
fitControl <- trainControl(## 10-fold CV
                           method = "repeatedcv",
                           number = 10,
                           ## repeated ten times
                           repeats = 10)
```

Current scope:

- *k*-fold cross-validation - "repeatedcv", "cv"
- leave-one-out cross-validation - "loocv"
- bootstrap - "boot"

Available Models in `caret` package

- http://topepo.github.io/caret/available-models.html

Linear Regression

- Models relationship between the magnitude of one variable in terms of others.

- R script

  lm.R

k-Nearest Neighbor

- ***k*-NN** is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification.
- The input consists of the *k* closest training examples in the feature space.

- R script

  knn.R

Decision Tree

- A **decision tree** is a model that uses a tree-like graph for prediction

- R script

  dt.R

Naive Bayes

- **Naive Bayes classifier** is a simple probabilistic classifier based on Bayes' theorem with strong (naive) independence assumptions between the features.

- R script

  nb.R

Neural Network

- **Artificial neural networks** (**ANNs**) are computing systems inspired by the biological neural networks that constitute animal brains.

- R script

  nnet.R

Testing

Data Splitting

```
set.seed(3456)
trainIndex <- createDataPartition(iris$Species, p = .8,
                                  list = FALSE)
head(trainIndex)

##      Resample1
## [1,]         1
## [2,]         2
## [3,]         4
## [4,]         5
## [5,]         6
## [6,]         8

irisTrain <- iris[ trainIndex,]
irisTest  <- iris[-trainIndex,]
```

Model Testing

```
predictions <- predict(modelFit, TestSet)

confusionMatrix(predictions, TestSet$Class)
```

Metrics

- Classification
  - Accuracy: Sum of *tp* and *tn* divided by total population
  - Kappa: Measures the agreement between two raters who each classify *N* items into *C* mutually exclusive categories
- Regression
  - RMSE (Root Mean Squared Error): The square root of the average squared error of the regression
  - R^2 (Coefficient of Determination, R squared): The proportion of variance explained by the model, from 0 to 1
  - MAE (Mean Absolute Error): The average magnitude of the errors in a set of predictions

Prediction

Prediction using trained model

```
predictions <- predict(modelFit, ValidationSet)
```

Thank You!

Questions?