**Log generated from Rattle**

set.seed(crv$seed)

crs$nobs <- nrow(crs$dataset) # 3183 observations

crs$sample <- crs$train <- sample(nrow(crs$dataset), 0.7\*crs$nobs) # 2228 observations

crs$validate <- sample(setdiff(seq\_len(nrow(crs$dataset)), crs$train), 0.15\*crs$nobs) # 477 observations

crs$test <- setdiff(setdiff(seq\_len(nrow(crs$dataset)), crs$train), crs$validate) # 478 observations

# The following variable selections have been noted.

crs$input <- c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar",

"chlorides", "free.sulfur.dioxide", "total.sulfur.dioxide", "density",

"pH", "sulphates", "alcohol", "quality")

crs$numeric <- c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar",

"chlorides", "free.sulfur.dioxide", "total.sulfur.dioxide", "density",

"pH", "sulphates", "alcohol", "quality")

crs$categoric <- NULL

crs$target <- NULL

crs$risk <- NULL

crs$ident <- NULL

crs$ignore <- NULL

crs$weights <- NULL

#============================================================

# Rattle timestamp: 2014-05-30 12:00:09 x86\_64-w64-mingw32

# Note the user selections.

# Build the training/validate/test datasets.

set.seed(crv$seed)

crs$nobs <- nrow(crs$dataset) # 3183 observations

crs$sample <- crs$train <- sample(nrow(crs$dataset), 0.7\*crs$nobs) # 2228 observations

crs$validate <- sample(setdiff(seq\_len(nrow(crs$dataset)), crs$train), 0.15\*crs$nobs) # 477 observations

crs$test <- setdiff(setdiff(seq\_len(nrow(crs$dataset)), crs$train), crs$validate) # 478 observations

# The following variable selections have been noted.

crs$input <- c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar",

"chlorides", "free.sulfur.dioxide", "total.sulfur.dioxide", "density",

"pH", "sulphates", "alcohol")

crs$numeric <- c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar",

"chlorides", "free.sulfur.dioxide", "total.sulfur.dioxide", "density",

"pH", "sulphates", "alcohol")

crs$categoric <- NULL

crs$target <- "quality"

crs$risk <- NULL

crs$ident <- NULL

crs$ignore <- NULL

crs$weights <- NULL

#============================================================

# Rattle timestamp: 2014-05-30 12:00:16 x86\_64-w64-mingw32

# Neural Network

# Build a neural network model using the nnet package.

require(nnet, quietly=TRUE)

# Build the NNet model.

set.seed(199)

crs$nnet <- nnet(quality ~ .,

data=crs$dataset[crs$sample,c(crs$input, crs$target)],

size=20, linout=TRUE, skip=TRUE, MaxNWts=10000, trace=FALSE, maxit=100)

# Print the results of the modelling.

cat(sprintf("A %s network with %d weights.\n",

paste(crs$nnet$n, collapse="-"),

length(crs$nnet$wts)))

cat(sprintf("Inputs: %s.\n",

paste(crs$nnet$coefnames, collapse=", ")))

cat(sprintf("Output: %s.\n",

names(attr(crs$nnet$terms, "dataClasses"))[1]))

cat(sprintf("Sum of Squares Residuals: %.4f.\n",

sum(residuals(crs$nnet) ^ 2)))

cat("\n")

print(summary(crs$nnet))

cat('\n')

# Time taken: 0.59 secs

#============================================================

# Rattle timestamp: 2014-05-30 12:00:20 x86\_64-w64-mingw32

# Evaluate model performance.

# NNET: Generate a Predicted v Observed plot for nnet model on app-prop.csv [validate].

crs$pr <- predict(crs$nnet, newdata=crs$dataset[crs$validate, c(crs$input, crs$target)])

# Obtain the observed output for the dataset.

obs <- subset(crs$dataset[crs$validate,], select=crs$target)

# Handle in case categoric target treated as numeric.

obs.rownames <- rownames(obs)

obs <- as.numeric(obs[[1]])

obs <- data.frame(quality=obs)

rownames(obs) <- obs.rownames

# Combine the observed values with the predicted.

fitpoints <- na.omit(cbind(obs, Predicted=crs$pr))

# Obtain the pseudo R2 - a correlation.

fitcorr <- format(cor(fitpoints[,1], fitpoints[,2])^2, digits=4)

# Plot settings for the true points and best fit.

op <- par(c(lty="solid", col="blue"))

# Display the observed (X) versus predicted (Y) points.

plot(jitter(fitpoints[[1]]), fitpoints[[2]], asp=1, xlab="quality (Jittered)", ylab="Predicted")

# Generate a simple linear fit between predicted and observed.

prline <- lm(fitpoints[,2] ~ fitpoints[,1])

# Add the linear fit to the plot.

abline(prline)

# Add a diagonal representing perfect correlation.

par(c(lty="dashed", col="black"))

abline(0, 1)

# Include a pseudo R-square on the plot

legend("bottomright", sprintf(" Pseudo R-square=%s ", fitcorr), bty="n")

# Add a title and grid to the plot.

title(main="Predicted vs. Observed

Neural Net Model

app-prop.csv [validate]",

sub=paste("Rattle", format(Sys.time(), "%Y-%b-%d %H:%M:%S"), Sys.info()["user"]))

grid()