kNN Clustering - Regression

Using 10-fold cross validations

Karen Mazidi

Load the data

```
library(ISLR)
df <- Auto[]
df$origin <- as.integer(df$origin)
# subset to columns mpg, weight, year, origin
df <- data.frame(scale(df[, c(1, 5, 7, 8)]  ))
```

Create the 10 folds

We could do this manually but there is a function in caret that does this. Since the Auto data is a little less than 400 rows, we expect each of the 10 folds to be of legth 40 or less. We confirm that with sapply.

```
library(caret)
```

```
## Loading required package: lattice

## Loading required package: ggplot2
```

```
set.seed(1234)
folds <- createFolds(df$mpg, k=10)
sapply(folds, length)
```

```
## Fold01 Fold02 Fold03 Fold04 Fold05 Fold06 Fold07 Fold08 Fold09 Fold10
##     39     40     40     38     40     40     38     39     39     39
```

Look at the fold indices

To get a better idea of the folds, let's just print the indices for each fold.

```
for (i in 1:10){
  print(folds[[i]])
}
```

```
##  [1]   2   7  26  45  52  59  62  73  78  80  85  95  98 109 118 147 209 220 249
## [20] 251 264 274 276 295 297 299 300 301 342 350 352 355 357 363 374 376 379 383
## [39] 384
##  [1]   5   6  15  56  71  88 103 107 119 121 122 129 131 134 135 139 141 165 175
## [20] 181 193 194 197 214 215 228 241 243 247 271 272 275 280 285 293 309 321 323
## [39] 330 373
##  [1]  13  25  29  36  43  46  66  74  86  92 117 127 130 140 146 157 159 164 167
## [20] 190 196 198 216 233 245 248 256 279 282 302 316 318 319 322 335 354 358 371
## [39] 380 381
##  [1]   4  14  18  23  28  34  35  50  75  96 102 133 142 143 154 158 169 177 201
## [20] 219 227 236 239 242 250 259 260 292 315 329 332 340 343 362 364 367 375 387
##  [1]  20  42  77  91  97 112 115 136 156 180 183 184 187 191 204 211 217 223 229
## [20] 234 237 240 244 246 257 258 262 268 281 287 289 298 304 314 333 336 349 356
## [39] 368 372
##  [1]   8   9  12  17  22  24  53  55  57  60  61  84  90  93 116 123 138 148 172
## [20] 189 192 202 208 224 231 255 263 278 283 286 294 317 337 339 347 351 359 370
## [39] 382 388
##  [1]  10  41  51  58  69  82  94 100 101 111 113 120 132 150 151 153 155 160 161
## [20] 170 174 178 185 205 206 212 225 254 270 277 324 326 328 341 361 366 378 391
##  [1]  21  38  47  48  49  54  64  65  67  76 104 105 106 110 114 144 149 162 171
## [20] 179 199 200 210 213 226 232 261 284 290 296 306 320 327 338 345 346 348 377
## [39] 390
##  [1]   1   3  11  16  19  27  32  40  63  81  83  89 108 128 137 145 152 163 166
## [20] 168 176 207 218 221 222 230 253 265 266 267 273 291 303 307 310 334 365 386
## [39] 389
##  [1]  30  31  33  37  39  44  68  70  72  79  87  99 124 125 126 173 182 186 188
## [20] 195 203 235 238 252 269 288 305 308 311 312 313 325 331 344 353 360 369 385
## [39] 392
```

Perform 10-fold cv

For now we will just let k=3 and perform 10-fold cv, then average the correlation and mse values.

```
test_mse <- rep(0, 10)
test_cor <- rep(0, 10)
for (i in 1:10){
  fit <- knnreg(df[-folds[[i]], 2:4], df$mpg[-folds[[i]]], k=3)
  pred <- predict(fit, df[folds[[i]], 2:4])
  test_cor[i] <- cor(pred, df$mpg[folds[[i]]])
  test_mse[i] <- mean((pred - df$mpg[folds[[i]]])^2)
}
print(paste("Average correlation is ", round(mean(test_cor), 2)))
```

```
## [1] "Average correlation is  0.93"
```

```
print(paste("range is ", range(test_cor)))
```

```
## [1] "range is  0.895630269179599" "range is  0.936723021049577"
```

```
print(paste("Average mse is ", round(mean(test_mse), 2)))
```

```
## [1] "Average mse is  0.16"
```

```
print(paste("range is ", range(test_mse)))
```

```
## [1] "range is  0.11930324928509"  "range is  0.268702111260869"
```

Try with various k

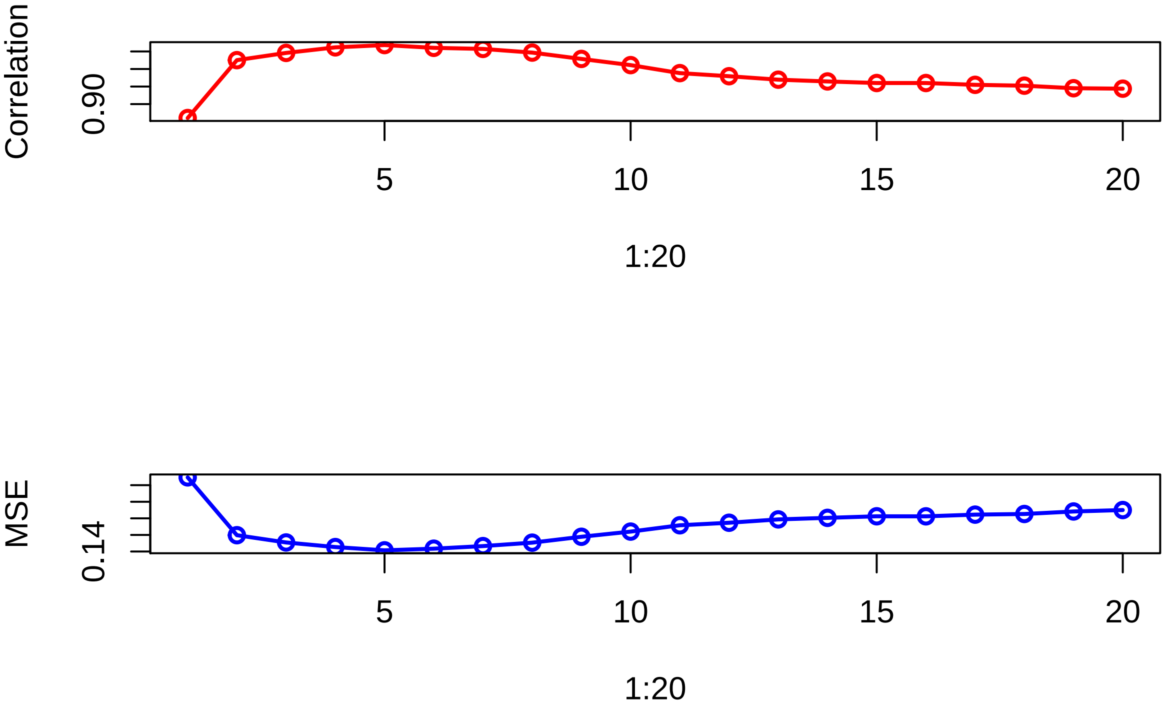
We modify the code above to be an anonymouse function called by sapply.

```
# try various values for k
k_values <- seq(1, 39, 2)
results <- sapply(k_values, function(k){
  mse_k <- rep(0, 10)
  cor_k <- rep(0, 10)
  for (i in 1:10){
    fit <- knnreg(df[-folds[[i]], 2:4], df$mpg[-folds[[i]]], k=k)
    pred <- predict(fit, df[folds[[i]], 2:4])
    cor_k[i] <- cor(pred, df$mpg[folds[[i]]])
    mse_k[i] <- mean((pred - df$mpg[folds[[i]]])^2)
  }
  #print(paste(mean(cor_k), mean(mse_k)))
  list(mean(cor_k), mean(mse_k))
})
# reshape results into matrix
m <- matrix(results, nrow=20, ncol=2, byrow=TRUE)
```

Examine results

Plot the correlation and mse for each value of k.

```r
par(mfrow=c(2, 1))
plot(1:20, unlist(m[,1]), lwd=2, type="o", col='red', ylab="Correlation")
plot(1:20, unlist(m[,2]), lwd=2, type="o", col='blue', ylab="MSE")
```


Correlation
1:20
MSE
1:20