Problem Set #2

Cian Stryker

10/1/2020

Conceptual Questions

Question 1

a.) The Euclidean distance between observation 3 and the test point is 1.732.

b.) Our prediction would be Black because the closest observation to the test point is observation 4, which is black.

c.) The three closest observations to the test point are observations 2, 4, and 5. Since observation 2 and 5 are white, our prediction with K = 3 would also be white. The majority of the closest three points are white.

d.) The prediction would be Y = 2 because the average of the three closest observations (2, 4, and 5) is 2.

Question 2

You would have 1,073,741,824 combinations with 30 potential covariates. This indicates that you would be better off using forward/backward selection instead of best subset selection. Trying to go with best subset selection would require you to try all 1 billion or so possible combinations, which would obviously take a ton of time. By using forward/backward selection you will save a lot of time and likely come very close to the accuracy of best subset selection, but it will be slightly worse.

Question 3 False

Question 4 I believe that I should side with Colleage C because this is a large p (predictor) and small n (observation) situation. KNN does not have a lot of data to actually work with here (only thirty observations) and therefore using KNN would likely lead to overfitting. Similarly, the large amount of predictors means assuming a linear relationship would be inappropriate. OLS methods do not work when p > n, so a linear regression would not be appropriate either. We should not use either KNN or linear regression in this situation.

Question 5 Colleague C may be concerned because QDA has to estimate a separate covariance matrix for each predictor in a model. In this example data we have 40,000 meaning using QDA would likely lead to very high variance.

Question 6 I would expect that logistic regression would outperform LDA because all our predictors are categorical and binary.

Question 7 The shrinkage penalty for ridge regression is called L2-norm, which is the sum of the squared coefficients.

Question 8 The shrinkage penalty for lasso regression is called L1-norm, which is the sum of the absolute coefficients.

Question 9 The main difference between lasso and ridge regression is that ridge regression will not perform feature selection, i.e. the number of predictors will not change. Lasso regression will actually shrink predictor coefficients to zero, which will remove them from the model. Lasso regression will therefore perform feature selection and remove predictors, which improves interpretability.

Data Questions

1.) One observation has missing values for at least one feature.

2.) There are five categorical variables in the data set: year, state, breath, jail, and service. State has 50 different classes, but breath, jail, and service each only have two classes.

3.) The adjusted R^2 value is 0.981

4.)

a.) The lambda with the lowest cross-validation error for 5 fold cross validation is 0.095.

b.) The cross validation error was 296.596.

c.) The standard error of the mean cross-validation error for this value of lambda was 47.12.

d.) The largest value of lamda whose mean cross-validation error was within one standard deviation of the lowest cross-validation is 0.807.

5.) The best k according to CV is 12.

```
set.seed(222)
cross_validation_KNN  <- function(data_x, data_y, k_seq, kfolds) {
  fold_ids      <- rep(seq(kfolds), ceiling(nrow(data_x) / kfolds))
  fold_ids      <- fold_ids[1:nrow(data_x)]
  fold_ids      <- sample(fold_ids, length(fold_ids))
  CV_error_mtx  <- matrix(0,
                          nrow = length(k_seq),
                          ncol = kfolds)
  for (k in k_seq) {
    for (fold in 1:kfolds) {

      knn_fold_model    <- knn(train = data_x[which(fold_ids != fold),],
                               test = data_x[which(fold_ids == fold),],
                               cl = data_y[which(fold_ids != fold)],
```

```
                                k = k)
      CV_error_mtx[k,fold]  <- mean(knn_fold_model !=
                                    data_y[which(fold_ids == fold)])
    }
  }
  return(CV_error_mtx)
}
```

6.)