ISYE 6501 Hw 10

Ryan Cherry

2024-03-24

Before performing any of the imputation methods, I ran a summary for each of the variables in the breast cancer data set to determine if there were any missing values and, if so, which variable contained them. After running the summary function, it was determined that the Bare Nuclei variable contained 16 total missing values, which needed to be dealt with.

```
#add specific variable names for each column

colnames(breast_cancer) <- c("ID", "Clump_Thickness",
"Uniform_Cell_Size", "Uniform_Cell_Shape", "Marg_Adhesion",
"Single_Epith_Cell_Size", "Bare_Nuclei", "Bland_Chromatin",
"Normal_Nucleoli", "Mitoses", "Class")

#run a summary of the breast cancer to see if there are any missing values
summary(breast_cancer)
```

```
##        ID           Clump_Thickness  Uniform_Cell_Size Uniform_Cell_Shape
##  Min.   :   61634   Min.   : 1.000   Min.   : 1.000    Min.   : 1.000
##  1st Qu.:  870688   1st Qu.: 2.000   1st Qu.: 1.000    1st Qu.: 1.000
##  Median : 1171710   Median : 4.000   Median : 1.000    Median : 1.000
##  Mean   : 1071704   Mean   : 4.418   Mean   : 3.134    Mean   : 3.207
##  3rd Qu.: 1238298   3rd Qu.: 6.000   3rd Qu.: 5.000    3rd Qu.: 5.000
##  Max.   :13454352   Max.   :10.000   Max.   :10.000    Max.   :10.000
##
##  Marg_Adhesion    Single_Epith_Cell_Size  Bare_Nuclei     Bland_Chromatin
##  Min.   : 1.000   Min.   : 1.000         Min.   : 1.000   Min.   : 1.000
##  1st Qu.: 1.000   1st Qu.: 2.000         1st Qu.: 1.000   1st Qu.: 2.000
##  Median : 1.000   Median : 2.000         Median : 1.000   Median : 3.000
##  Mean   : 2.807   Mean   : 3.216         Mean   : 3.545   Mean   : 3.438
##  3rd Qu.: 4.000   3rd Qu.: 4.000         3rd Qu.: 6.000   3rd Qu.: 5.000
##  Max.   :10.000   Max.   :10.000         Max.   :10.000   Max.   :10.000
##                                          NA's   :16
##  Normal_Nucleoli     Mitoses           Class
##  Min.   : 1.000   Min.   : 1.000   Min.   :2.00
##  1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.:2.00
##  Median : 1.000   Median : 1.000   Median :2.00
##  Mean   : 2.867   Mean   : 1.589   Mean   :2.69
##  3rd Qu.: 4.000   3rd Qu.: 1.000   3rd Qu.:4.00
##  Max.   :10.000   Max.   :10.000   Max.   :4.00
##
```

Question 14.1

1. Use the mean/mode imputation method to impute values for the missing data.

The specific observations that had missing data were determined with a combination of the which function and the is.Na command. However, before imputation could be performed, we have to check that no more than 5% of the observations have missing values, which is a recommended threshold in industry. In this case, 16 observations were missing from the breast cancer data set, which was only 2.28% of the data. This did not exceed the threshold of 55, so imputation could be performed on this data set.

After this check, the observations without missing values were separated from the ones with missing data in order to perform four types of imputation.

```
#determine whether the threshold for imputation is met
percent_missing = (16/699) * 100
percent_missing
```

```
## [1] 2.288984
```

```
#isolate the missing observations in preparation for imputation
missing_observations<- which(is.na(breast_cancer$Bare_Nuclei))
missing_observations
```

```
##  [1]  24  41 140 146 159 165 236 250 276 293 295 298 316 322 412 618
```

```
non_missing <- breast_cancer[-missing_observations, ]

missing <- breast_cancer[missing_observations, ]
```

To perform mean imputation, I took the average of the bare nuclei values, with the missing observations excluded. The average turned out to be 3.55, which is closer to the upper tail of the distribution of values, considering the median is 1. As a result, the mean imputation may place too much influence on the missing observations. We are potentially introducing some bias or error, since each of the imputed values will have the same value, which would be unlikely in reality. The new values are also not reflective of the full range of values shown by the rest of the single nuclei values. This means that the predictive power of the resulting model is likely reduced.

```
#use mean imputation on the bare nuclei variable
average <- mean(non_missing$Bare_Nuclei)
average
```

```
## [1] 3.544656
```

```
#replace Na values with the average of the non-missing values
mean_bare_nuclei <- breast_cancer$Bare_Nuclei
mean_bare_nuclei[is.na(breast_cancer$Bare_Nuclei)] <- average

mean_imputed <- breast_cancer
mean_imputed$Bare_Nuclei <- mean_bare_nuclei
head(mean_imputed)
```

```
##        ID Clump_Thickness Uniform_Cell_Size Uniform_Cell_Shape Marg_Adhesion
## 1 1000025               5                 1                  1             1
## 2 1002945               5                 4                  4             5
## 3 1015425               3                 1                  1             1
## 4 1016277               6                 8                  8             1
## 5 1017023               4                 1                  1             3
## 6 1017122               8                10                 10             8
##   Single_Epith_Cell_Size Bare_Nuclei Bland_Chromatin Normal_Nucleoli Mitoses
## 1                      2           1               3               1       1
## 2                      7          10               3               2       1
## 3                      2           2               3               1       1
## 4                      3           4               3               7       1
## 5                      2           1               3               1       1
## 6                      7          10               9               7       1
##   Class
## 1     2
## 2     2
## 3     2
## 4     2
## 5     2
## 6     4
```

In addition to mean imputation, mode imputation was performed on the bare nuclei variable. The mode of the existing observations was 1, so missing data points were replaced with a value of 1. However, we may not know the story behind the missing data points and without that information, too much weight may be placed on the missing observations. The mode imputation results in values in the middle 50% of the data, but no randomness is added which means that we are potentially introducing bias. As above, even though the mode-imputed values are lower than the mean imputed values, each imputed value is the same and is therefore not reflective of the spread of the values for single nuclei. This means that the predictive power of any resulting model may be reduced.

```r
#get the mode of the bare nuclei variable
modes <- Mode(non_missing$Bare_Nuclei)
modes
```

```
## [1] 1
## attr(,"freq")
## [1] 402
```

```r
#replace Na values in bare nuclei variables with the mode of the non-missing
#observations
mode_bare_nuclei <- breast_cancer$Bare_Nuclei
mode_bare_nuclei[is.na(breast_cancer$Bare_Nuclei)] <- modes

mode_imputed <- breast_cancer
mode_imputed$Bare_Nuclei <- mode_bare_nuclei
head(mode_imputed)
```

```
##        ID Clump_Thickness Uniform_Cell_Size Uniform_Cell_Shape Marg_Adhesion
## 1 1000025               5                 1                  1             1
## 2 1002945               5                 4                  4             5
## 3 1015425               3                 1                  1             1
## 4 1016277               6                 8                  8             1
## 5 1017023               4                 1                  1             3
## 6 1017122               8                10                 10             8
##   Single_Epith_Cell_Size Bare_Nuclei Bland_Chromatin Normal_Nucleoli Mitoses
## 1                      2           1               3               1       1
## 2                      7          10               3               2       1
## 3                      2           2               3               1       1
## 4                      3           4               3               7       1
## 5                      2           1               3               1       1
## 6                      7          10               9               7       1
##   Class
## 1     2
## 2     2
## 3     2
## 4     2
## 5     2
## 6     4
```

2. Use regression to impute values for the missing data.

Regression imputation was also used to fill in the missing values in the single nuclei variable. A regression model was built using single nuclei, the variable with the missing data, as the response and the other variables as predictors. The final regression model is then used to predict the values for the missing observations, which will take on a wider range of values, as shown in the output below. In order to match the whole numbers shown in the rest of the values for the single nuclei variable, the predicted values were rounded to the nearest whole integer.

```
#create a regression model with each predictor regressing onto single nuclei,
#the variable with the missing observations
model <- lm(Bare_Nuclei ~ Clump_Thickness + Uniform_Cell_Size + Uniform_Cell_Shape
+ Marg_Adhesion + Single_Epith_Cell_Size + Bland_Chromatin + Normal_Nucleoli +
Mitoses, data = non_missing)

#predict the values for all 16 missing single nuclei observations using the
#regression model created above
single_nuclei_regression_prediction <- round(predict(model, missing))
single_nuclei_regression_prediction
```

```
##  24  41 140 146 159 165 236 250 276 293 295 298 316 322 412 618
##   5   8   1   2   1   2   3   2   2   6   1   2   6   2   1   1
```

The resulting values predicted y the regression model better follow the distribution of the non-missing observations. As a result, there is lower weight placed on several of the missing observations due to higher randomness. More randomness is introduced to the single nuclei variables, which means that the resulting models may have slightly higher predictive power or accuracy than ones made using mean or mode-imputed values. However,

```
#replace Na values in bare nuclei variable with regressed values
regression_prediction <- breast_cancer$Bare_Nuclei
regression_prediction[is.na(breast_cancer$Bare_Nuclei)] <- single_nuclei_regression_prediction

regression <- breast_cancer
regression$Bare_Nuclei <- regression_prediction
head(regression)
```

```
##        ID Clump_Thickness Uniform_Cell_Size Uniform_Cell_Shape Marg_Adhesion
## 1 1000025               5                 1                  1             1
## 2 1002945               5                 4                  4             5
## 3 1015425               3                 1                  1             1
## 4 1016277               6                 8                  8             1
## 5 1017023               4                 1                  1             3
## 6 1017122               8                10                 10             8
##   Single_Epith_Cell_Size Bare_Nuclei Bland_Chromatin Normal_Nucleoli Mitoses
## 1                      2           1               3               1       1
## 2                      7          10               3               2       1
## 3                      2           2               3               1       1
## 4                      3           4               3               7       1
## 5                      2           1               3               1       1
## 6                      7          10               9               7       1
##   Class
## 1     2
## 2     2
## 3     2
## 4     2
## 5     2
## 6     4
```

3. Use regression with perturbation to impute values for the missing data.

The final imputation method used on the breast cancer data set involves both regression and perturbation. The process of perturbation involved using the rnorm function to generate generate 16 values that reflect a random normal distribution with a mean of 0 and standard deviation of 1. These generated values were then added to the predicted values from the regression model created above to get the final imputed values. As with the regression imputation, the values were rounded to the nearest whole integer to better reflect the non-missing observation values.

```r
set.seed(123)

#use rnorm to create random variables with characteristics of a standard normal
#distribution
perturb <- abs(rnorm(length(missing), mean = 0, sd = 1))

#add newly created random variables to predicted values from regression model in
#question 2
perturbation_bare_nuclei <- round(predict(model, missing) + perturb)
```

```
## Warning in predict(model, missing) + perturb: longer object length is not a
## multiple of shorter object length
```

```r
perturbation_bare_nuclei
```

```
##  24  41 140 146 159 165 236 250 276 293 295 298 316 322 412 618
##   6   8   2   2   1   4   3   3   3   6   2   3   6   3   1   1
```

From the output above, regression with perturbation produced a range of values that better reflect a normal distribution. This places less weight on the missing observations and can result in a model with higher predictive power. More randomness is introduced to the single nuclei variable to get a range of values more reflective of the non-imputed values for single nuclei. As a result, the predictive power of the resulting model will be higher.

```r
#replace Na values in bare nuclei variable with perturbed values
reg_and_perturb_bare_nuclei <- breast_cancer$Bare_Nuclei
reg_and_perturb_bare_nuclei[is.na(breast_cancer$Bare_Nuclei)] <-
perturbation_bare_nuclei

perturbation <- breast_cancer
perturbation$Bare_Nuclei <- reg_and_perturb_bare_nuclei
head(perturbation)
```

```
##        ID Clump_Thickness Uniform_Cell_Size Uniform_Cell_Shape Marg_Adhesion
## 1 1000025               5                 1                  1             1
## 2 1002945               5                 4                  4             5
## 3 1015425               3                 1                  1             1
## 4 1016277               6                 8                  8             1
## 5 1017023               4                 1                  1             3
## 6 1017122               8                10                 10             8
##   Single_Epith_Cell_Size Bare_Nuclei Bland_Chromatin Normal_Nucleoli Mitoses
## 1                      2           1               3               1       1
## 2                      7          10               3               2       1
## 3                      2           2               3               1       1
## 4                      3           4               3               7       1
## 5                      2           1               3               1       1
## 6                      7          10               9               7       1
##   Class
## 1     2
## 2     2
## 3     2
## 4     2
## 5     2
## 6     4
```

Overall, regression with perturbation would be the technique I would recommend for imputing missing values. The resulting randomness introduced to the data are higher than what is introduced by regression imputation and can give higher predictive power and accuracy to any resulting model. In addition, with perturbation, we do not have to just use a normal distribution to generate data values like we did here. If we can find out the actual distribution of the values for the predictor in question, we can use the same distribution to generate random values that match the variability of the non-missing observations.

Question 15.1

Describe a situation or problem from your job, everyday life, current events, etc., for which optimization would be appropriate. What data would you need?

Optimization would be appropriate in the functional area of supply chain, specifically logistics, where we seek to minimize the cost of transport of raw materials or parts for a certain product. The objective function would be to minimize transportation costs and the constraints are the total transport time, money the company has available, and many more. The key decisions (variables) are the transport mode, political events that may impact transport, environmental conditions, customer demand, and many others.

Optimizing transport time at the lowest possible cost is a key consideration faced by supply chain professionals on a daily basis.