Question 14.1

1. Use the mean/mode imputation method to impute values for the missing data.
2. Use regression to impute values for the missing data.
3. Use regression with perturbation to impute values for the missing data.
4. (Optional) Compare the results and quality of classification models (e.g., SVM, KNN) build using

   (1) the data sets from questions 1,2,3;
   (2) the data that remains after data points with missing values are removed; and (3) the data set when a binary variable is introduced to indicate missing values.

Clear global environment, load and preview dataset. Summary of the dataset suggests only V7 contains 16 missing values, with missing value proportion being 16 over 699 obs or 0.0229. Since missing value proportion is well below .05 threshold, therefore I proceed to perform mean/mode imputation.

```
> rm(list=ls())
> df1 <- read.table("/Users/henryyang/Desktop/Gatech/SP20/ISYE6501/HW10/breast-cancer-wisconsin.data.txt", header = F, stringsAsFactors = F, na.strings = "?", sep = ",")
> head(df1,3)
       V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11
1 1000025  5  1  1  1  2  1  3  1   1   2
2 1002945  5  4  4  5  7 10  3  2   1   2
3 1015425  3  1  1  1  2  2  3  1   1   2
> summary(df1)
       V1                V2               V3
 Min.   :   61634   Min.   : 1.000   Min.   : 1.000
 1st Qu.:  870688   1st Qu.: 2.000   1st Qu.: 1.000
 Median : 1171710   Median : 4.000   Median : 1.000
 Mean   : 1071704   Mean   : 4.418   Mean   : 3.134
 3rd Qu.: 1238298   3rd Qu.: 6.000   3rd Qu.: 5.000
 Max.   :13454352   Max.   :10.000   Max.   :10.000

       V4               V5               V6               V7
 Min.   : 1.000   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
 1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.: 2.000   1st Qu.: 1.000
 Median : 1.000   Median : 1.000   Median : 2.000   Median : 1.000
 Mean   : 3.207   Mean   : 2.807   Mean   : 3.216   Mean   : 3.545
 3rd Qu.: 5.000   3rd Qu.: 4.000   3rd Qu.: 4.000   3rd Qu.: 6.000
 Max.   :10.000   Max.   :10.000   Max.   :10.000   Max.   :10.000
                                                     NA's   :16
       V8               V9              V10              V11
 Min.   : 1.000   Min.   : 1.000   Min.   : 1.000   Min.   :2.00
 1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.:2.00
 Median : 3.000   Median : 1.000   Median : 1.000   Median :2.00
 Mean   : 3.438   Mean   : 2.867   Mean   : 1.589   Mean   :2.69
 3rd Qu.: 5.000   3rd Qu.: 4.000   3rd Qu.: 1.000   3rd Qu.:4.00
 Max.   :10.000   Max.   :10.000   Max.   :10.000   Max.   :4.00
```

Impute the mean/mode for the missing values

```
> df_mean <- df1
```

```
> df_mean$V7[is.na(df_mean$V7)] <- mean(df_mean$V7, na.rm = TRUE)
> mean(df_mean$V7)
[1] 3.544656
> df_mode <- df1
> df_mode$V7[is.na(df_mode$V7)] <- mode(df_mode$V7)
```

Use regression to impute values for the missing data. Linear regression model is constructed using data without na values and response variables. Summary of the model suggests some variables are not significant.

```
> dfdrop <- na.omit(df1)
> model1 <- lm(V7~V2+V3+V4+V5+V6+V8+V9+V10, data = dfdrop)
> summary(model1)

Call:
lm(formula = V7 ~ V2 + V3 + V4 + V5 + V6 + V8 + V9 + V10, data = dfdrop)

Residuals:
   Min      1Q  Median      3Q     Max
-9.7316 -0.9426 -0.3002  0.6725  8.6998

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.616652   0.194975  -3.163  0.00163 **
V2          0.230156   0.041691   5.521 4.83e-08 ***
V3         -0.067980   0.076170  -0.892  0.37246
V4          0.340442   0.073420   4.637 4.25e-06 ***
V5          0.339705   0.045919   7.398 4.13e-13 ***
V6          0.090392   0.062541   1.445  0.14883
V8          0.320577   0.059047   5.429 7.91e-08 ***
V9          0.007293   0.044486   0.164  0.86983
V10        -0.075230   0.059331  -1.268  0.20524
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.274 on 674 degrees of freedom
Multiple R-squared:  0.615,	Adjusted R-squared:  0.6104
F-statistic: 134.6 on 8 and 674 DF,  p-value: < 2.2e-16
```

Linear regression model is re-fitted by including variables that are significant. Summary of the model suggests all variables are significant.

```
> step(model1, trace = 0)

Call:
lm(formula = V7 ~ V2 + V4 + V5 + V8, data = dfdrop)

Coefficients:
(Intercept)           V2           V4           V5           V8
    -0.5360       0.2262       0.3173       0.3323       0.3238

> model2 <- lm(V7~V2+V4+V5+V8, data=dfdrop)
```

```
> summary(model2)

Call:
lm(formula = V7 ~ V2 + V4 + V5 + V8, data = dfdrop)

Residuals:
    Min      1Q  Median      3Q     Max 
-9.8115 -0.9531 -0.3111  0.6678  8.6889 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -0.53601    0.17514  -3.060   0.0023 ** 
V2           0.22617    0.04121   5.488 5.75e-08 ***
V4           0.31729    0.05086   6.239 7.76e-10 ***
V5           0.33227    0.04431   7.499 2.03e-13 ***
V8           0.32378    0.05606   5.775 1.17e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.274 on 678 degrees of freedom
Multiple R-squared:  0.6129,	Adjusted R-squared:  0.6107 
F-statistic: 268.4 on 4 and 678 DF,  p-value: < 2.2e-16

# Impute missing values to predicted values from the previous regression model.
> pred <- predict(model2, data = df1[is.na(df1$V7)])
> df_reg <- df1
> df_reg$V7[is.na(df_reg$V7)] <- pred

# Use regression with perturbation to impute values for the missing data with mice package.
> library(mice)
> pert <- mice(df1, method = "norm.nob", m = 1)

iter imp variable
  1   1  V7
  2   1  V7
  3   1  V7
  4   1  V7
  5   1  V7
> df_pert <- complete(pert)
```

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?

In my job, use optimization to allocate the amount of certain parts spent on the assembly of products would be ideal to maximize profits using available resources in the inventory. In order to achieve allocation optimization, we would need data such as parts quantity in inventory, daily production quantity objectives, daily production capacity to provide relevant inputs to the optimization model.