Armando Tejeda

Homework Week 10

11/2/22

Question 14.1

The breast cancer data set breast-cancer-wisconsin.data.txt from http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/ (description at http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29) has missing values.

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.

Solution:

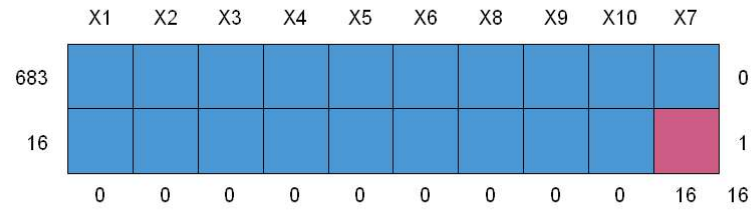
In [ ]:
```
# load packages I may need

library(tidyverse)
library(caret)
library(modelr)
library(ggthemes)
library(corrplot)

library(mice)
library(VIM)
```

In [ ]:
```
set.seed(1)

# read in the cancer data
cancer_df = read_delim('C:/Users/ateje/OneDrive/Desktop/VS Code Projects/GTx_MM_in_Ana
                       col_names = F, na = c('?')) %>%
        as.data.frame() %>%
        mutate(X11 = ifelse(X11 == 2, 'Benign', 'Malignant'))


# using mice to see missing data - including complete cases
md.pattern(cancer_df[,-11])

# plot the missing data
plot_missing <- aggr(cancer_df, col = c('navyblue', 'red'), numbers = T, sortVars = T)
```

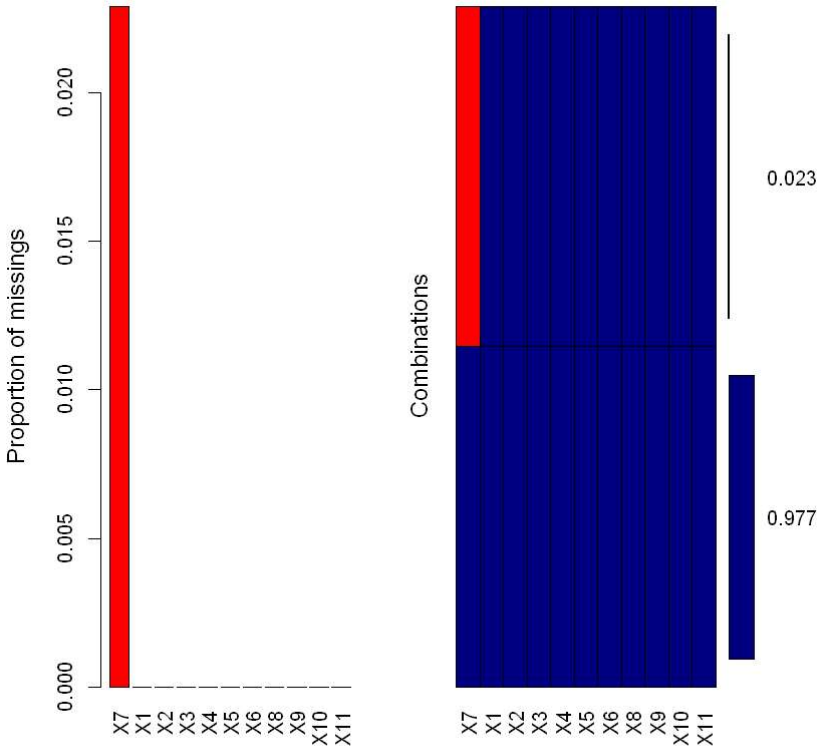
```
Rows: 699 Columns: 11
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
dbl (11): X1, X2, X3, X4, X5, X6, X7, X8, X9, X10, X11

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

A matrix: 3 × 11 of type dbl

	X1	X2	X3	X4	X5	X6	X8	X9	X10	X7	
683	1	1	1	1	1	1	1	1	1	1	0
16	1	1	1	1	1	1	1	1	1	0	1
	0	0	0	0	0	0	0	0	0	16	16




```
Variables sorted by number of missings:
Variable      Count
      X7 0.02288984
      X1 0.00000000
      X2 0.00000000
      X3 0.00000000
      X4 0.00000000
      X5 0.00000000
      X6 0.00000000
      X8 0.00000000
      X9 0.00000000
     X10 0.00000000
     X11 0.00000000
```



In [ ]:
```
# imputation using the mice package - mean imputation
mean_impute <- mice(cancer_df, m = 5, meth = 'mean' )

# look at the values
mean_impute$imp
```

```
 iter imp variable
  1   1  X7
  1   2  X7
  1   3  X7
  1   4  X7
  1   5  X7
  2   1  X7
  2   2  X7
  2   3  X7
  2   4  X7
  2   5  X7
  3   1  X7
  3   2  X7
  3   3  X7
  3   4  X7
  3   5  X7
  4   1  X7
  4   2  X7
  4   3  X7
  4   4  X7
  4   5  X7
  5   1  X7
  5   2  X7
  5   3  X7
  5   4  X7
  5   5  X7
```

```
Warning message:
"Number of logged events: 1"
```

$X1

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X2

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X3

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X4

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X5

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X6

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X7

A data.frame: 16 × 5

	1	2	3	4	5
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
24	3.544656	3.544656	3.544656	3.544656	3.544656
41	3.544656	3.544656	3.544656	3.544656	3.544656
140	3.544656	3.544656	3.544656	3.544656	3.544656
146	3.544656	3.544656	3.544656	3.544656	3.544656
159	3.544656	3.544656	3.544656	3.544656	3.544656
165	3.544656	3.544656	3.544656	3.544656	3.544656
236	3.544656	3.544656	3.544656	3.544656	3.544656
250	3.544656	3.544656	3.544656	3.544656	3.544656
276	3.544656	3.544656	3.544656	3.544656	3.544656
293	3.544656	3.544656	3.544656	3.544656	3.544656
295	3.544656	3.544656	3.544656	3.544656	3.544656
298	3.544656	3.544656	3.544656	3.544656	3.544656
316	3.544656	3.544656	3.544656	3.544656	3.544656

	1	2	3	4	5
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
322	3.544656	3.544656	3.544656	3.544656	3.544656
412	3.544656	3.544656	3.544656	3.544656	3.544656
618	3.544656	3.544656	3.544656	3.544656	3.544656

$X8

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X9

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X10

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X11

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

```
 iter imp variable
  1   1  X7
  1   2  X7
  1   3  X7
  1   4  X7
  1   5  X7
  2   1  X7
  2   2  X7
  2   3  X7
  2   4  X7
  2   5  X7
  3   1  X7
  3   2  X7
  3   3  X7
  3   4  X7
  3   5  X7
  4   1  X7
  4   2  X7
  4   3  X7
  4   4  X7
  4   5  X7
  5   1  X7
  5   2  X7
  5   3  X7
  5   4  X7
  5   5  X7
```

```
Warning message:
"Number of logged events: 1"
```

$X1

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X2

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X3

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X4

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X5

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X6

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X7

A data.frame: 16 × 5

	1	2	3	4	5
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
24	5.3669508	5.3669508	5.3669508	5.3669508	5.3669508
41	8.1907122	8.1907122	8.1907122	8.1907122	8.1907122
140	0.8738591	0.8738591	0.8738591	0.8738591	0.8738591
146	1.6463893	1.6463893	1.6463893	1.6463893	1.6463893
159	1.0731978	1.0731978	1.0731978	1.0731978	1.0731978
165	2.1870186	2.1870186	2.1870186	2.1870186	2.1870186
236	2.7459168	2.7459168	2.7459168	2.7459168	2.7459168
250	2.0127161	2.0127161	2.0127161	2.0127161	2.0127161
276	2.3072038	2.3072038	2.3072038	2.3072038	2.3072038
293	5.9989744	5.9989744	5.9989744	5.9989744	5.9989744
295	1.1204527	1.1204527	1.1204527	1.1204527	1.1204527
298	2.6839366	2.6839366	2.6839366	2.6839366	2.6839366
316	5.6353059	5.6353059	5.6353059	5.6353059	5.6353059

	1	2	3	4	5
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
322	1.8585015	1.8585015	1.8585015	1.8585015	1.8585015
412	0.8587684	0.8587684	0.8587684	0.8587684	0.8587684
618	0.5907393	0.5907393	0.5907393	0.5907393	0.5907393

$X8

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X9

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X10

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X11

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

In [ ]:
```r
# imputation using the mice package - regression ignoring model error
regression_impute <- mice(cancer_df, m = 5, meth = 'norm.predict')

# look at the values
regression_impute$imp
```

In [ ]:
```r
# imputation using the mice package - perturbation impute
pert_impute <- mice(cancer_df, m = 5, meth = 'norm.nob')


# look at the values
pert_impute$imp
```

```
 iter imp variable
  1   1  X7
  1   2  X7
  1   3  X7
  1   4  X7
  1   5  X7
  2   1  X7
  2   2  X7
  2   3  X7
  2   4  X7
  2   5  X7
  3   1  X7
  3   2  X7
  3   3  X7
  3   4  X7
  3   5  X7
  4   1  X7
  4   2  X7
  4   3  X7
  4   4  X7
  4   5  X7
  5   1  X7
  5   2  X7
  5   3  X7
  5   4  X7
  5   5  X7
```

```
Warning message:
"Number of logged events: 1"
```

$X1

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X2

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X3

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X4

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X5

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X6

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X7

A data.frame: 16 × 5

	1	2	3	4	5
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
24	3.74019613	2.112548	5.754193	6.542245038	8.3991056
41	6.47785909	13.327107	11.031538	5.279079278	7.3691419
140	1.02816124	-1.205451	2.338500	-0.401999456	0.3160545
146	1.48513771	-2.179527	3.562301	0.008079856	-0.1830189
159	2.33526892	-1.725926	-1.833393	-0.559918789	0.8977351
165	1.73465992	-1.528415	1.617340	-0.988170624	1.5126888
236	3.11101934	1.408588	-2.822924	6.147719083	5.4472161
250	0.09159326	2.701089	3.150801	-1.743670278	3.4485212
276	2.38663866	-1.583771	3.171041	2.070783192	2.4976321
293	6.07586930	5.927439	3.048709	6.976790384	6.6228084
295	1.98902076	1.424256	3.886388	3.508502749	3.8107415
298	2.63590012	2.728575	4.621931	4.930007558	-1.4773002
316	5.16059793	4.842153	6.895866	4.993188229	3.4155540

	1	2	3	4	5
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
322	1.09375193	2.803795	2.331938	4.507931165	-1.8924155
412	-0.38016150	3.449344	-2.296806	1.278841938	0.2325643
618	0.88052310	-3.112713	1.746031	-0.125077946	-0.8736359

$X8

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X9

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X10

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

$X11

A data.frame: 0 × 5

1	2	3	4	5
<lgl>	<lgl>	<lgl>	<lgl>	<lgl>

In [ ]:
```r
# Data with mean impute
cancer_mean_df <- complete(mean_impute)
cancer_mean_df
```

A data.frame: 699 × 11

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1000025	5	1	1	1	2	1.000000	3	1	1	Benign
1002945	5	4	4	5	7	10.000000	3	2	1	Benign
1015425	3	1	1	1	2	2.000000	3	1	1	Benign
1016277	6	8	8	1	3	4.000000	3	7	1	Benign
1017023	4	1	1	3	2	1.000000	3	1	1	Benign
1017122	8	10	10	8	7	10.000000	9	7	1	Malignant
1018099	1	1	1	1	2	10.000000	3	1	1	Benign
1018561	2	1	2	1	2	1.000000	3	1	1	Benign
1033078	2	1	1	1	2	1.000000	1	1	5	Benign
1033078	4	2	1	1	2	1.000000	2	1	1	Benign
1035283	1	1	1	1	1	1.000000	3	1	1	Benign
1036172	2	1	1	1	2	1.000000	2	1	1	Benign
1041801	5	3	3	3	2	3.000000	4	4	1	Malignant
1043999	1	1	1	1	2	3.000000	3	1	1	Benign
1044572	8	7	5	10	7	9.000000	5	5	4	Malignant
1047630	7	4	6	4	6	1.000000	4	3	1	Malignant
1048672	4	1	1	1	2	1.000000	2	1	1	Benign
1049815	4	1	1	1	2	1.000000	3	1	1	Benign
1050670	10	7	7	6	4	10.000000	4	1	2	Malignant
1050718	6	1	1	1	2	1.000000	3	1	1	Benign
1054590	7	3	2	10	5	10.000000	5	4	4	Malignant
1054593	10	5	5	3	6	7.000000	7	10	1	Malignant
1056784	3	1	1	1	2	1.000000	2	1	1	Benign
1057013	8	4	5	1	2	3.544656	7	3	1	Malignant
1059552	1	1	1	1	2	1.000000	3	1	1	Benign
1065726	5	2	3	4	2	7.000000	3	6	1	Malignant
1066373	3	2	1	1	1	1.000000	2	1	1	Benign
1066979	5	1	1	1	2	1.000000	2	1	1	Benign
1067444	2	1	1	1	2	1.000000	2	1	1	Benign
1070935	1	1	3	1	2	1.000000	1	1	1	Benign
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1350423	5	10	10	8	5	5	7	10	1	Malignant

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1352848	3	10	7	8	5	8	7	4	1	Malignant
1353092	3	2	1	2	2	1	3	1	1	Benign
1354840	2	1	1	1	2	1	3	1	1	Benign
1354840	5	3	2	1	3	1	1	1	1	Benign
1355260	1	1	1	1	2	1	2	1	1	Benign
1365075	4	1	4	1	2	1	1	1	1	Benign
1365328	1	1	2	1	2	1	2	1	1	Benign
1368267	5	1	1	1	2	1	1	1	1	Benign
1368273	1	1	1	1	2	1	1	1	1	Benign
1368882	2	1	1	1	2	1	1	1	1	Benign
1369821	10	10	10	10	5	10	10	10	7	Malignant
1371026	5	10	10	10	4	10	5	6	3	Malignant
1371920	5	1	1	1	2	1	3	2	1	Benign
466906	1	1	1	1	2	1	1	1	1	Benign
466906	1	1	1	1	2	1	1	1	1	Benign
534555	1	1	1	1	2	1	1	1	1	Benign
536708	1	1	1	1	2	1	1	1	1	Benign
566346	3	1	1	1	2	1	2	3	1	Benign
603148	4	1	1	1	2	1	1	1	1	Benign
654546	1	1	1	1	2	1	1	1	8	Benign
654546	1	1	1	3	2	1	1	1	1	Benign
695091	5	10	10	5	4	5	4	4	1	Malignant
714039	3	1	1	1	2	1	1	1	1	Benign
763235	3	1	1	1	2	1	2	1	2	Benign
776715	3	1	1	1	3	2	1	1	1	Benign
841769	2	1	1	1	2	1	1	1	1	Benign
888820	5	10	10	3	7	3	8	10	2	Malignant
897471	4	8	6	4	3	4	10	6	1	Malignant
897471	4	8	8	5	4	5	10	4	1	Malignant

A data.frame: 699 × 11

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1000025	5	1	1	1	2	1.000000	3	1	1	Benign
1002945	5	4	4	5	7	10.000000	3	2	1	Benign
1015425	3	1	1	1	2	2.000000	3	1	1	Benign
1016277	6	8	8	1	3	4.000000	3	7	1	Benign
1017023	4	1	1	3	2	1.000000	3	1	1	Benign
1017122	8	10	10	8	7	10.000000	9	7	1	Malignant
1018099	1	1	1	1	2	10.000000	3	1	1	Benign
1018561	2	1	2	1	2	1.000000	3	1	1	Benign
1033078	2	1	1	1	2	1.000000	1	1	5	Benign
1033078	4	2	1	1	2	1.000000	2	1	1	Benign
1035283	1	1	1	1	1	1.000000	3	1	1	Benign
1036172	2	1	1	1	2	1.000000	2	1	1	Benign
1041801	5	3	3	3	2	3.000000	4	4	1	Malignant
1043999	1	1	1	1	2	3.000000	3	1	1	Benign
1044572	8	7	5	10	7	9.000000	5	5	4	Malignant
1047630	7	4	6	4	6	1.000000	4	3	1	Malignant
1048672	4	1	1	1	2	1.000000	2	1	1	Benign
1049815	4	1	1	1	2	1.000000	3	1	1	Benign
1050670	10	7	7	6	4	10.000000	4	1	2	Malignant
1050718	6	1	1	1	2	1.000000	3	1	1	Benign
1054590	7	3	2	10	5	10.000000	5	4	4	Malignant
1054593	10	5	5	3	6	7.000000	7	10	1	Malignant
1056784	3	1	1	1	2	1.000000	2	1	1	Benign
1057013	8	4	5	1	2	5.366951	7	3	1	Malignant
1059552	1	1	1	1	2	1.000000	3	1	1	Benign
1065726	5	2	3	4	2	7.000000	3	6	1	Malignant
1066373	3	2	1	1	1	1.000000	2	1	1	Benign
1066979	5	1	1	1	2	1.000000	2	1	1	Benign
1067444	2	1	1	1	2	1.000000	2	1	1	Benign
1070935	1	1	3	1	2	1.000000	1	1	1	Benign
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1350423	5	10	10	8	5	5	7	10	1	Malignant

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1352848	3	10	7	8	5	8	7	4	1	Malignant
1353092	3	2	1	2	2	1	3	1	1	Benign
1354840	2	1	1	1	2	1	3	1	1	Benign
1354840	5	3	2	1	3	1	1	1	1	Benign
1355260	1	1	1	1	2	1	2	1	1	Benign
1365075	4	1	4	1	2	1	1	1	1	Benign
1365328	1	1	2	1	2	1	2	1	1	Benign
1368267	5	1	1	1	2	1	1	1	1	Benign
1368273	1	1	1	1	2	1	1	1	1	Benign
1368882	2	1	1	1	2	1	1	1	1	Benign
1369821	10	10	10	10	5	10	10	10	7	Malignant
1371026	5	10	10	10	4	10	5	6	3	Malignant
1371920	5	1	1	1	2	1	3	2	1	Benign
466906	1	1	1	1	2	1	1	1	1	Benign
466906	1	1	1	1	2	1	1	1	1	Benign
534555	1	1	1	1	2	1	1	1	1	Benign
536708	1	1	1	1	2	1	1	1	1	Benign
566346	3	1	1	1	2	1	2	3	1	Benign
603148	4	1	1	1	2	1	1	1	1	Benign
654546	1	1	1	1	2	1	1	1	8	Benign
654546	1	1	1	3	2	1	1	1	1	Benign
695091	5	10	10	5	4	5	4	4	1	Malignant
714039	3	1	1	1	2	1	1	1	1	Benign
763235	3	1	1	1	2	1	2	1	2	Benign
776715	3	1	1	1	3	2	1	1	1	Benign
841769	2	1	1	1	2	1	1	1	1	Benign
888820	5	10	10	3	7	3	8	10	2	Malignant
897471	4	8	6	4	3	4	10	6	1	Malignant
897471	4	8	8	5	4	5	10	4	1	Malignant

A data.frame: 699 × 11

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1000025	5	1	1	1	2	1.000000	3	1	1	Benign
1002945	5	4	4	5	7	10.000000	3	2	1	Benign
1015425	3	1	1	1	2	2.000000	3	1	1	Benign
1016277	6	8	8	1	3	4.000000	3	7	1	Benign
1017023	4	1	1	3	2	1.000000	3	1	1	Benign
1017122	8	10	10	8	7	10.000000	9	7	1	Malignant
1018099	1	1	1	1	2	10.000000	3	1	1	Benign
1018561	2	1	2	1	2	1.000000	3	1	1	Benign
1033078	2	1	1	1	2	1.000000	1	1	5	Benign
1033078	4	2	1	1	2	1.000000	2	1	1	Benign
1035283	1	1	1	1	1	1.000000	3	1	1	Benign
1036172	2	1	1	1	2	1.000000	2	1	1	Benign
1041801	5	3	3	3	2	3.000000	4	4	1	Malignant
1043999	1	1	1	1	2	3.000000	3	1	1	Benign
1044572	8	7	5	10	7	9.000000	5	5	4	Malignant
1047630	7	4	6	4	6	1.000000	4	3	1	Malignant
1048672	4	1	1	1	2	1.000000	2	1	1	Benign
1049815	4	1	1	1	2	1.000000	3	1	1	Benign
1050670	10	7	7	6	4	10.000000	4	1	2	Malignant
1050718	6	1	1	1	2	1.000000	3	1	1	Benign
1054590	7	3	2	10	5	10.000000	5	4	4	Malignant
1054593	10	5	5	3	6	7.000000	7	10	1	Malignant
1056784	3	1	1	1	2	1.000000	2	1	1	Benign
1057013	8	4	5	1	2	3.740196	7	3	1	Malignant
1059552	1	1	1	1	2	1.000000	3	1	1	Benign
1065726	5	2	3	4	2	7.000000	3	6	1	Malignant
1066373	3	2	1	1	1	1.000000	2	1	1	Benign
1066979	5	1	1	1	2	1.000000	2	1	1	Benign
1067444	2	1	1	1	2	1.000000	2	1	1	Benign
1070935	1	1	3	1	2	1.000000	1	1	1	Benign
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1350423	5	10	10	8	5	5	7	10	1	Malignant

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1352848	3	10	7	8	5	8	7	4	1	Malignant
1353092	3	2	1	2	2	1	3	1	1	Benign
1354840	2	1	1	1	2	1	3	1	1	Benign
1354840	5	3	2	1	3	1	1	1	1	Benign
1355260	1	1	1	1	2	1	2	1	1	Benign
1365075	4	1	4	1	2	1	1	1	1	Benign
1365328	1	1	2	1	2	1	2	1	1	Benign
1368267	5	1	1	1	2	1	1	1	1	Benign
1368273	1	1	1	1	2	1	1	1	1	Benign
1368882	2	1	1	1	2	1	1	1	1	Benign
1369821	10	10	10	10	5	10	10	10	7	Malignant
1371026	5	10	10	10	4	10	5	6	3	Malignant
1371920	5	1	1	1	2	1	3	2	1	Benign
466906	1	1	1	1	2	1	1	1	1	Benign
466906	1	1	1	1	2	1	1	1	1	Benign
534555	1	1	1	1	2	1	1	1	1	Benign
536708	1	1	1	1	2	1	1	1	1	Benign
566346	3	1	1	1	2	1	2	3	1	Benign
603148	4	1	1	1	2	1	1	1	1	Benign
654546	1	1	1	1	2	1	1	1	8	Benign
654546	1	1	1	3	2	1	1	1	1	Benign
695091	5	10	10	5	4	5	4	4	1	Malignant
714039	3	1	1	1	2	1	1	1	1	Benign
763235	3	1	1	1	2	1	2	1	2	Benign
776715	3	1	1	1	3	2	1	1	1	Benign
841769	2	1	1	1	2	1	1	1	1	Benign
888820	5	10	10	3	7	3	8	10	2	Malignant
897471	4	8	6	4	3	4	10	6	1	Malignant
897471	4	8	8	5	4	5	10	4	1	Malignant

In [ ]:
```
# Data with regression prediction impute
cancer_regression_df <- complete(regression_impute)
cancer_regression_df
```

In [ ]:
```
# Data with pertubation impute
```

```
cancer_pert_df <- complete(pert_impute)
cancer_pert_df
```

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?

Solution:

Optimization can be used for airline to decide the number of first-class tickets, advantage tickets, and coach tickets they should sell to maximize their profits for their flights. The company might need to consider certain constraints like number of first class and coach seats available and the size of the staff to serve first class, etc.