431 Project Study 2 Demonstration

Thomas E. Love

2017-12-10

Contents

1 Setup in R

```
library(pander); library(mice); library(Epi)
library(gridExtra); library(vcd); library(Hmisc)
library(mosaic); library(car); library(forcats)
library(tidyverse)

source("Love-boost.R")

hbp_study <- read.csv("hbp_study.csv") %>% tbl_df
```

2 What is this?

This document demonstrates analyses needed for Task E of your project Study 2 (using your data.)

To fix ideas, we will use simulated data from a study of high blood pressure in 999 African-American adult subjects who are not of Hispanic or Latino ethnicity. To be included, the subject had to be between 33 and 83 years of age at baseline, have a series of items available in their health record at baseline, including a baseline systolic blood pressure, and then return for a blood pressure check 18 months later. Our goal will be to build a prediction model for the subject's *change* in systolic blood pressure over the 18-month period, on the basis of some of their characteristics at baseline.

The data (which, again, are simulated), are in the `hbp_study.csv` data file on the Projects - Your Data page of our website.

2.1 Revised Instructions

This document makes use of the revised instructions for Study 2 (Task E) found in the Project Instructions after the Proposal document posted to our website on the evening of November 21, 2016. Those revised instructions are repeated in the steps that follow.

3 The Original Data Set and Range Checks/Missingness (Project Task D)

The `hbp_study` data set includes 12 variables and 999 adult subjects. For each subject, we have gathered

- baseline information on their `age`, and their `sex`,
- whether or not they have a `diabetes` diagnosis,
- the socio-economic status of their neighborhood of residence (`nses`),
- their body-mass index (`bmi1`) and systolic blood pressure (`sbp1`),
- their `insurance` type, `tobacco` use history, and
- whether or not they have a prescription for a `statin`, or for a `diuretic`.
- Eighteen months later, we gathered a new systolic blood pressure (`sbp2`) for each subject.

```
glimpse(hbp_study)
```

```
Observations: 999
Variables: 12
$ id        <fctr> A0001, A0004, A0005, A0013, A0015, A0017, A0018, A0...
$ age       <int> 58, 65, 61, 51, 61, 45, 40, 50, 43, 46, 56, 52, 58, ...
$ sex       <fctr> F, F, F, M, F, F, F, F, M, F, F, F, M, F, M, F, F, ...
```

```
$ diabetes  <fctr> No, No, Yes, No, No, No, Yes, Yes, No, No, No, No, ...
$ nses      <fctr> Low, Very Low, Very Low, Very Low, Very Low, Low, V...
$ bmi1      <dbl> 24.41, 50.50, 29.76, 41.83, 30.95, 33.01, 36.32, 30....
$ sbp1      <int> 147, 134, 170, 118, 132, 110, 127, 152, 125, 161, 14...
$ insurance <fctr> Medicaid, Medicaid, Medicaid, Medicaid, Medicaid, M...
$ tobacco   <fctr> never, never, current, quit, never, current, never,...
$ statin    <int> 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0...
$ diuretic  <int> 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1...
$ sbp2      <int> 138, 134, 140, 143, 162, 141, 101, 154, 111, 154, 15...
```

This tibble describes twelve variables, including:

- a categorical `id` variable not to be used in our model except for identification of subjects,
- two variables that, when combined, make up our outcome (`sbp1` and `sbp2`),
- seven categorical candidate predictors, specifically `sex`, `diabetes`, `nses`, `insurance`, `tobacco`, `statin`, and `diuretic`
- three quantitative candidate predictors, specifically `age`, `bmi1` and `sbp1`.

3.1 Which variables should be included in the tidy data set?

Note that I'm not planning to use all of these predictors in my models, but I'm going to build a tidy data set including all of them anyway, so I can demonstrate solutions to some problems you might have. When you build your tidy data set, restrict it to the variables (outcomes, predictors and id) that you will actually use in your modeling.

4 Data Management: Building a Tidy Data Set (Project Task D)

In building our tidy version of these data, we must:

- calculate and store the outcome variable (`sbp_diff` = `sbp2 - sbp1`),
- deal with the ordering of levels in the multi-categorical variables `nses`, `insurance` and `tobacco`,
- change the name of `nses` to something more helpful - I'll use `nbhd_ses` as the new name[1].

4.1 Dealing with Missingness

Note that you will need to ensure that any *missing* values are appropriately specified using `NA`.

- In this data set, we're all set on that issue.
  - There are missing data in `nses` (8 NA), `bmi1` (5 NA) and `tobacco` (23 NA).
  - In these data, we will eventually have to deal with the missing data in a rational way, but we'll do that *after* building the tidy data set and codebook.
- [**Missing Outcomes**] Your tidy data set should also delete any subjects with missing values of your outcome variable.
  - The elements (`sbp1` and `sbp2`) that go into our outcome, `sbp_diff`, have no missing values, though, so we'll be OK in that regard.

In building the tidy data set, leave all missing values for candidate predictors as `NA`.

4.2 Calculating the `sbp_diff` outcome

The simplest approach to creating the new difference and storing it in `hbp_study` follows:

[1] Admittedly, that's not much better.

```
hbp_study$sbp_diff <- hbp_study$sbp2 - hbp_study$sbp1
Hmisc::describe(hbp_study$sbp_diff)
```

```
hbp_study$sbp_diff
       n  missing distinct     Info     Mean      Gmd      .05      .10
     999        0      110        1   -2.776    23.27      -37      -29
     .25      .50      .75      .90      .95
     -17       -2       10       23       30

lowest : -60 -58 -57 -56 -54, highest:  50  52  54  56  57
```

We have no missing values in our outcome, and each of the values look plausible. Some subjects had large changes in their systolic blood pressure from baseline to follow-up, as large as a 60 mm Hg difference, it appears. The average change across our 999 subjects was modest at about 2 mm Hg, which seems reasonable, and none of the individual values seem unreasonable[2], so we'll move on.

4.3 Re-ordering the levels of the categorical variables

For categorical variables, it's always worth it to check to see whether the existing orders of the factor levels match the inherent order of the information.

```
levels(hbp_study$nses)
```

```
[1] "High"     "Low"      "Middle"   "Very Low"
```

```
levels(hbp_study$tobacco)
```

```
[1] "current" "never"   "quit"
```

```
levels(hbp_study$insurance)
```

```
[1] "Medicaid"  "Medicare"  "Private"   "Uninsured"
```

- The order of `nses`, instead of the alphabetical ("High", "Low", "Middle", "Very Low"), should go from "Very Low" to "Low" to "Middle" to "High", or perhaps its reverse.
- For `tobacco`, instead of ("current", "never", "quit"), we want ("never", "quit", "current").
- For `insurance`, we'll change the order to ("Medicare", "Private", "Medicaid", "Uninsured")

Let's fix that using the `fct_relevel` function from the `forcats` package.

```
hbp_study$nses <- fct_relevel(hbp_study$nses, "Very Low", "Low", "Middle", "High")
hbp_study$tobacco <- fct_relevel(hbp_study$tobacco, "never", "quit", "current")
hbp_study$insurance <- fct_relevel(hbp_study$insurance, "Medicare", "Private",
                                   "Medicaid", "Uninsured")
```

We'll also reorder the `diabetes` variable to put "Yes" before "No".

```
hbp_study$diabetes <- fct_relevel(hbp_study$diabetes, "Yes")
```

Note that any levels left out of a `fct_relevel` statement get included in their current order, after whatever levels have been specified.

[2] A change of 60 mm Hg in systolic blood pressure in 18 months is certainly unusual, but in 999 patients, we can't be that surprised to see a change that extreme, especially since we see several other people with similar changes in the data.

4.4 Change the name of `nses` to `nbhd_ses`

We can simply create the new variable, using `hbp_study$nbhd_ses <- hbp_study$nses` and then remove the `nses` variable from our final data set, but I'll use `dplyr` to rename the variable.

```
hbp_study <- dplyr::rename(hbp_study, nbhd_ses = nses)
```

4.5 Cleaning Up to get to our final data set

Let's build a data set, called `hbp_tidy` that contains only the twelve variables in our code book.

```
hbp_tidy <- select(hbp_study, id, sbp_diff, sbp1, age, sex,
                   diabetes, nbhd_ses, bmi1, insurance,
                   tobacco, statin, diuretic )
Hmisc::describe(hbp_tidy)
```

```
hbp_tidy

 12  Variables      999  Observations
--------------------------------------------------------------------------------
id
       n  missing distinct
     999        0      999

lowest : A0001 A0002 A0003 A0004 A0005, highest: A0995 A0996 A0997 A0998 A0999
--------------------------------------------------------------------------------
sbp_diff
       n  missing distinct     Info     Mean      Gmd      .05      .10
     999        0      110        1   -2.776    23.27      -37      -29
     .25      .50      .75      .90      .95
     -17       -2       10       23       30

lowest : -60 -58 -57 -56 -54, highest:  50  52  54  56  57
--------------------------------------------------------------------------------
sbp1
       n  missing distinct     Info     Mean      Gmd      .05      .10
     999        0      101        1    136.5    20.39    108.9    115.0
     .25      .50      .75      .90      .95
   124.0    136.0    147.0    160.0    168.0

lowest :  81  83  91  92  93, highest: 198 201 202 203 205
--------------------------------------------------------------------------------
age
       n  missing distinct     Info     Mean      Gmd      .05      .10
     999        0       51    0.999    58.69    11.93     41.0     45.0
     .25      .50      .75      .90      .95
    52.0     59.0     66.0     73.2     76.0

lowest : 33 34 35 36 37, highest: 79 80 81 82 83
--------------------------------------------------------------------------------
sex
       n  missing distinct
     999        0        2
```

```
Value          F     M
Frequency    655   344
Proportion 0.656 0.344
```

```
diabetes
       n  missing distinct
     999        0        2

Value        Yes    No
Frequency    331   668
Proportion 0.331 0.669
```

```
nbhd_ses
       n  missing distinct
     991        8        4

Value      Very Low      Low   Middle     High
Frequency       220      336      281      154
Proportion    0.222    0.339    0.284    0.155
```

```
bmi1
       n  missing distinct     Info     Mean      Gmd      .05      .10
     994        5      834        1    33.72    9.139    22.66    24.40
     .25      .50      .75      .90      .95
   27.86    32.14    38.36    45.08    49.96

lowest : 16.72 17.79 18.00 18.44 18.54, highest: 64.30 65.43 65.46 65.95 74.65
```

```
insurance
       n  missing distinct
     999        0        4

Value       Medicare   Private  Medicaid Uninsured
Frequency        402       160       398        39
Proportion     0.402     0.160     0.398     0.039
```

```
tobacco
       n  missing distinct
     976       23        3

Value        never    quit current
Frequency      319     362     295
Proportion   0.327   0.371   0.302
```

```
statin
       n  missing distinct     Info      Sum     Mean      Gmd
     999        0        2     0.74      556   0.5566   0.4941
```

```
diuretic
       n  missing distinct     Info      Sum     Mean      Gmd
     999        0        2    0.668      665   0.6657   0.4456
```
---

5 The Codebook (Project Task D)

The 12 variables in our tidy data set for this demonstration are as follows.

Variable	Type	Description / Levels
id	Categorical	subject code (A001-A999)
sbp_diff	Quantitative	outcome variable, SBP after 18 months minus SBP at baseline, in mm Hg
sbp1	Quantitative	baseline SBP (systolic blood pressure), in mm Hg
age	Quantitative	age of subject at baseline, in years
sex	Binary	Male or Female
diabetes	Binary	Does subject have a diabetes diagnosis: Yes or No
nbhd_ses	4 level Cat.	Socio-economic status of subject's home neighborhood: Very Low, Low, Middle and High
bmi1	Quantitative	subject's body-mass index at baseline
insurance	4 level Cat.	subject's insurance status at baseline: Medicare, Private, Medicaid, Uninsured
tobacco	3 level Cat.	subject's tobacco use at baseline: never, quit (former), current
statin	Binary	1 = statin prescription at baseline, else 0
diuretic	Binary	1 = diuretic prescription at baseline, else 0

6 Step 0. Work for Project Task E on Missing Values

6.1 Revised Instructions

Identify all the variables in your tidy data set that have missing (NA) values. Delete all observations with missing outcomes, and use simple imputation to impute values for the candidate predictors with NAs. Use the resulting imputed data set in all subsequent work.

6.2 Identifying Missing Values

We can use the `md.pattern` function from the `mice` package.

```
md.pattern(hbp_tidy)
```

```
    id sbp_diff sbp1 age sex diabetes insurance statin diuretic bmi1
963  1        1    1   1   1        1         1      1        1    1
  8  1        1    1   1   1        1         1      1        1    1
  5  1        1    1   1   1        1         1      1        1    0
 23  1        1    1   1   1        1         1      1        1    1
     0        0    0   0   0        0         0      0        0    5
    nbhd_ses tobacco
963        1       1  0
  8        0       1  1
  5        1       1  1
 23        1       0  1
           8      23 36
```

Or, the `colSums` approach gives a count of `NA` values by column in the data frame.

```
colSums(is.na(hbp_tidy))
```

```
       id  sbp_diff      sbp1       age       sex  diabetes  nbhd_ses
        0         0         0         0         0         0         8
     bmi1 insurance   tobacco    statin  diuretic
        5         0        23         0         0
```

We have 963 subjects with no missing values, 8 who are missing `nbhd_ses`, another 5 who are missing `bmi1` and 23 who are missing `tobacco`.

6.3 A Note on the Models I will use

In this example, I have been working with a large set of candidate predictor variables, so that I can demonstrate some data management issues.

In what follows, I will restrict myself to the following five predictors: `sbp1`, `age`, `bmi1`, `diabetes`, and `tobacco`, in trying to predict `sbp_diff`.

To that end, I'll create a new data set, called `hbp_small` which includes only the `id` value, the outcome `sbp_diff` and these five predictors.

```
hbp_small <- select(hbp_tidy, id, sbp_diff, sbp1, age, bmi1, diabetes, tobacco)
```

6.4 Building Simple Imputations for Predictors with NAs

In no way am I suggesting this is good practice outside of this project, but for now, we'll do a simple imputation to fill in values for the missing `tobacco` and `bmi1` values, creating a new data frame which is completed for our subsequent work.

```
hbp_temp <- mice(hbp_small, m = 1, maxit = 5, method = 'pmm', seed = 431001)
```

```
 iter imp variable
  1   1  bmi1  tobacco
  2   1  bmi1  tobacco
  3   1  bmi1  tobacco
  4   1  bmi1  tobacco
  5   1  bmi1  tobacco
```

Note: If this approach bombs out for you, try these three things, in this order.

1. Save your work, close down R and R Studio, and then re-open them and try again, but this time, use `maxit = 1` rather than `maxit = 5`.
2. If that doesn't work, try `method = 'sample'` instead. Changing `method` to `sample` imputes with a random sample from the existing observations for each variable.
3. If even that doesn't work, delete the subjects with missing values using the `filter` command as discussed in the Project Instructions after Proposal about deleting rows with missing outcomes (section 7) and then press on with your new, smaller data set.

Once we have the imputed data, we then complete the data set to fill in the missing values:

```
hbp_s <- mice::complete(hbp_temp, 1)
```

This may take a moment or two, but when it's finished, the resulting `hbp_s` will have no missing values.

```
colSums(is.na(hbp_s))
```

```
      id sbp_diff      sbp1       age      bmi1 diabetes  tobacco
       0        0         0         0         0        0        0
```

7 Step 1. Develop training and test samples.

7.1 Revised Instructions

Obtain a training sample with a randomly selected 80% of your data, and have the remaining 20% in a test sample, properly labeled, and using `set.seed` so that the results can be replicated later.

7.2 R code

I'll create a training sample, with 80% of the data, called `hbp_s_training` and a test sample, with the remaining 20% of the data, called `hbp_s_test`.

```
set.seed(431123) # set your own seed, don't use this one
hbp_s_training <- hbp_s %>% sample_frac(.80)
hbp_s_test <- anti_join(hbp_s, hbp_s_training, by = "id")
dim(hbp_s) # number of rows and columns in hbp_s
```

```
[1] 999   7
```

```
dim(hbp_s_training) # check to be sure we have 80% of hbp_s here
```

```
[1] 799   7
```

```
dim(hbp_s_test) # check to be sure we have the rest of hbp_s here
```

```
[1] 200   7
```

8 Step 2. Summarize outcome and predictors numerically and assess the outcome's distribution graphically.

8.1 Revised Instructions

Using the training sample, provide numerical summaries of each predictor variable and the outcome (with Hmisc::describe), as well as graphical summaries of the outcome variable. Your results should now show no missing values in any variable. Are there any evident problems, such as substantial skew in the outcome variable?

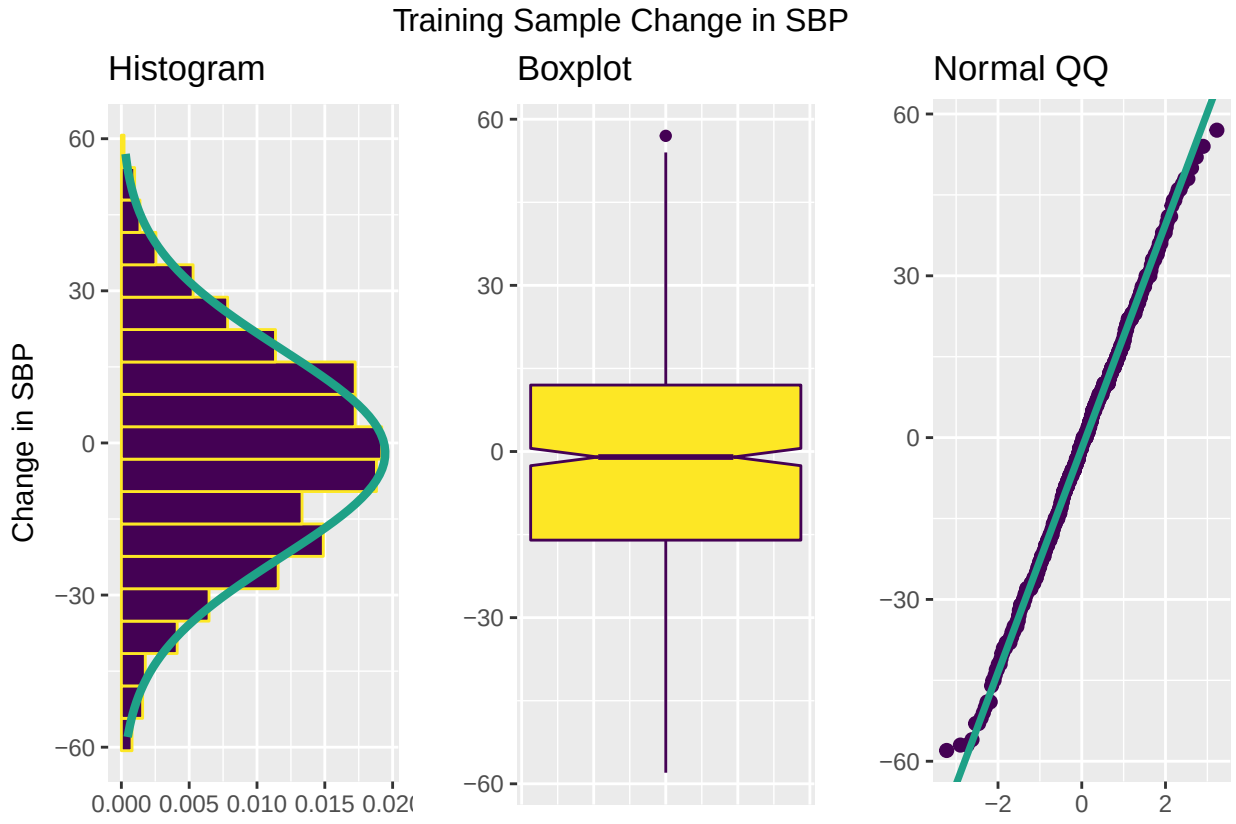
8.2 R code

```
Hmisc::describe(hbp_s_training)
```

```
hbp_s_training

 7  Variables      799  Observations
--------------------------------------------------------------------------------
id
       n  missing distinct
```

```
     799        0      799

lowest : A0001 A0002 A0003 A0006 A0007, highest: A0994 A0995 A0996 A0997 A0999
---------------------------------------------------------------------------
sbp_diff
       n  missing distinct     Info     Mean      Gmd      .05      .10
     799        0      106        1   -2.003    23.24      -36      -28
     .25      .50      .75      .90      .95
     -16       -1       12       24       31

lowest : -58 -57 -56 -53 -52, highest:  48  50  52  54  57
---------------------------------------------------------------------------
sbp1
       n  missing distinct     Info     Mean      Gmd      .05      .10
     799        0       95        1    136.4    20.37    109.0    115.0
     .25      .50      .75      .90      .95
   124.0    136.0    147.0    159.2    168.0

lowest :  81  83  91  92  93, highest: 190 193 198 202 203
---------------------------------------------------------------------------
age
       n  missing distinct     Info     Mean      Gmd      .05      .10
     799        0       50    0.999    58.66    11.94       41       45
     .25      .50      .75      .90      .95
      52       59       66       73       76

lowest : 33 34 35 36 37, highest: 78 79 80 81 82
---------------------------------------------------------------------------
bmi1
       n  missing distinct     Info     Mean      Gmd      .05      .10
     799        0      696        1    33.48    9.041    22.09    24.29
     .25      .50      .75      .90      .95
   27.79    32.13    38.28    44.15    49.25

lowest : 16.72 17.79 18.00 18.44 18.54, highest: 62.28 65.43 65.46 65.95 74.65
---------------------------------------------------------------------------
diabetes
       n  missing distinct
     799        0        2

Value       Yes   No
Frequency   248  551
Proportion 0.31 0.69
---------------------------------------------------------------------------
tobacco
       n  missing distinct
     799        0        3

Value        never    quit current
Frequency      259     305     235
Proportion   0.324   0.382   0.294
---------------------------------------------------------------------------
```

```
eda.1sam(dataframe = hbp_s_training,
         variable = hbp_s_training$sbp_diff,
         x.title = "Change in SBP",
         ov.title = "Training Sample Change in SBP")
```



I see no problems with a Normal model for the outcomes in this case.

9 Step 3. Build and interpret scatterplot matrix; consider potential transformations of your outcome.

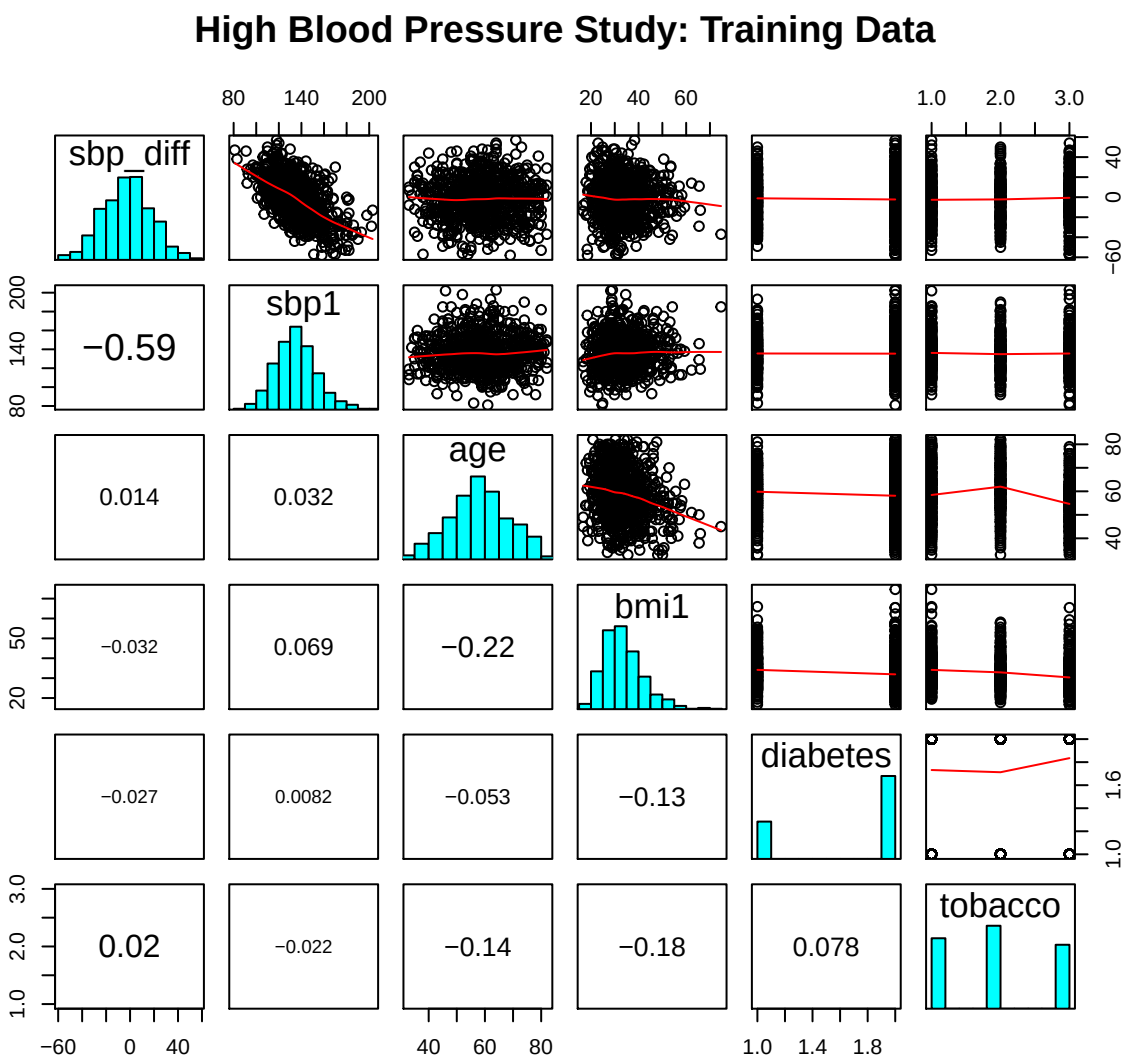
9.1 Revised Instructions

- Build and interpret a scatterplot matrix to describe the associations (both numerically and graphically) between the outcome and all predictors.
- Use a Box-Cox plot to investigate whether a transformation of your outcome is suggested.
- Describe what a correlation matrix suggests about collinearity between candidate predictors.

9.2 R Code

```
pairs (~ sbp_diff + sbp1 + age + bmi1 + diabetes + tobacco,
       data=hbp_s_training,
       main="High Blood Pressure Study: Training Data",
```

```
      upper.panel = panel.smooth,
      diag.panel = panel.hist,
      lower.panel = panel.cor)
```



9.2.1 Collinearity Checking

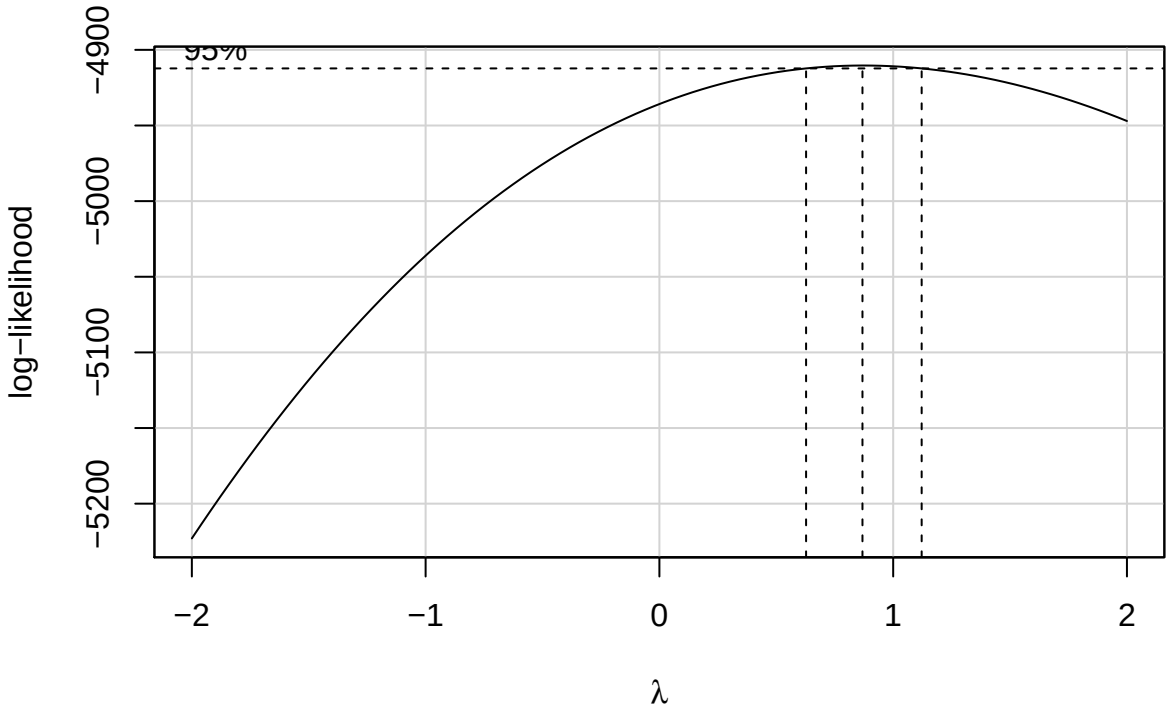
As for collinearity, none of these candidate predictors show any substantial correlation with each other. The largest Pearson correlation (in absolute value) between predictors is (-0.22) for `age` and `bmi1`, and that's not strong. As we'll see in Step 4, none of the generalized variance inflation factors exceed 1.2, let alone the 5 or so that we'd have to see to be seriously concerned about collinearity.

9.2.2 boxCox function to assess need for transformation of our outcome

To use the `boxCox` approach here, we need to realize that the distribution of our outcome, `sbp_diff`, includes negative values as well as zeros. The smallest `sbp_diff` value is -60. We'll need to add a value to each

`sbp_diff` in order to run the boxCox plot, so that the resulting "outcome" is strictly positive. I'll add 100. Although we're generally using a 90% confidence interval in this project, we won't worry about that issue in the `boxCox` plot, and instead just look at the point estimate from `powerTransform`.

```
boxCox(lm((sbp_diff + 100) ~ sbp1 + age + bmi1 + diabetes + tobacco, data = hbp_s_training))
```




```
powerTransform(lm((sbp_diff + 100) ~ sbp1 + age + bmi1 + diabetes + tobacco, data = hbp_s_training))
```

```
Estimated transformation parameters
       Y1
0.8726675
```

The estimated power transformation is about 0.9, and that's closer to 1 (the raw data) than any of the other transformations I'd consider from Tukey's ladder, so I won't apply a transformation[3].

10 Step 4. Build "kitchen sink" model, and describe/assess it.

10.1 Revised Instructions

Specify a "kitchen sink" linear regression model to describe the relationship between your outcome (potentially after transformation) and the main effects of each of your predictors.

[3]If your outcome data are substantially multimodal, I wouldn't look at the boxCox results as meaningful. Otherwise, it is up to you to decide whether a transformation suggested by boxCox should be applied to your data. Don't make the transformation if you wouldn't be able to interpret the result well, which probably means you should stick to transformations of strictly positive outcomes, and to the square root, square, logarithm and inverse transformations. If you do decide to include a transformation of your outcome in fitting models, be sure to back-transform any predictions you make at the end of the study (in Step 7), so that we can understand the prediction error results.

- Assess the overall effectiveness, within your training sample, of your model, by specifying and interpreting the R^2, adjusted R^2 (especially in light of your collinearity conclusions below), the residual standard error, and the ANOVA F test.
- Does collinearity in the kitchen sink model have a meaningful impact? How can you tell?
- Specify the size, magnitude and meaning of all coefficients, and identify appropriate conclusions regarding effect sizes with 90% confidence intervals.

10.2 R Code

```
mod.ksink <- lm(sbp_diff ~ sbp1 + age + bmi1 + diabetes + tobacco, data = hbp_s_training)
mod.ksink
```

```

Call:
lm(formula = sbp_diff ~ sbp1 + age + bmi1 + diabetes + tobacco,
    data = hbp_s_training)

Coefficients:
(Intercept)         sbp1          age         bmi1    diabetes2
   81.99546     -0.66526      0.09801      0.05424     -0.93695
   tobacco2     tobacco3
   -1.34195      1.08822
```

Our model predicts the `sbp_diff` using the predictors `sbp1`, `age`, `bmi1`, `diabetes` and `tobacco`.

```
summary(mod.ksink)
```

```

Call:
lm(formula = sbp_diff ~ sbp1 + age + bmi1 + diabetes + tobacco,
    data = hbp_s_training)

Residuals:
    Min      1Q  Median      3Q     Max
-45.741 -11.238  -0.154   9.715  53.420

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 81.99546    6.58405  12.454   <2e-16 ***
sbp1        -0.66526    0.03231 -20.587   <2e-16 ***
age          0.09801    0.06060   1.617    0.106
bmi1         0.05424    0.07587   0.715    0.475
diabetes2   -0.93695    1.28748  -0.728    0.467
tobacco2    -1.34195    1.41839  -0.946    0.344
tobacco3     1.08822    1.55319   0.701    0.484
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 16.59 on 792 degrees of freedom
Multiple R-squared:  0.3502,    Adjusted R-squared:  0.3453
F-statistic: 71.14 on 6 and 792 DF,  p-value: < 2.2e-16
```

Assess the overall effectiveness, within your training sample, of your model, by specifying and interpreting the R^2, adjusted R^2 (especially in light of your collinearity conclusions below), the residual standard error, and the ANOVA F test.

- This model accounts for just over 35% of the variation in `sbp_diff` in our training sample of 799 subjects.
- The adjusted R^2 (0.345) is very close to the raw R^2 (0.350), suggesting that we're not likely to have a serious problem with collinearity.
- The residual standard error is about 16.5 mm Hg, which indicates that about 95% of our subjects in this training sample should have model predictions within 33 mm Hg of the actual value of their `sbp_diff`, and nearly all should be within 49.5 mm Hg. Based on the maximum and minimum residuals, and a sample of 799 observations, it looks like there might be an outlier on the high end (a residual of 53.4), but on the low end, things look reasonable.
- The ANOVA F test p value (which is zero for all reasonable purposes) indicates a highly statistically significant amount of predictive value is accounted for by the model. This is no surprise given the moderate R^2 value and reasonably large ($n = 799$) size of this training sample.

Does collinearity in the kitchen sink model have a meaningful impact? How can you tell?

```
car::vif(mod.ksink)
```

```
             GVIF Df GVIF^(1/(2*Df))
sbp1     1.010595  1        1.005284
age      1.170928  1        1.082094
bmi1     1.138214  1        1.066871
diabetes 1.029679  1        1.014731
tobacco  1.154209  2        1.036504
```

No, it doesn't. We'd need to see a generalized variance inflation factor above 5 for collinearity to be a meaningful concern.

Specify the size, magnitude and meaning of all coefficients, and identify appropriate conclusions regarding effect sizes with 90% confidence intervals.

```
summary(mod.ksink)$coefficients
```

```
               Estimate Std. Error     t value     Pr(>|t|)
(Intercept) 81.99546053 6.58404767  12.4536554 1.219510e-32
sbp1        -0.66525940 0.03231401 -20.5873335 9.217418e-76
age          0.09800552 0.06059549   1.6173733 1.061959e-01
bmi1         0.05423503 0.07586525   0.7148863 4.748899e-01
diabetes2   -0.93695295 1.28747987  -0.7277418 4.669867e-01
tobacco2    -1.34194560 1.41838704  -0.9461068 3.443827e-01
tobacco3     1.08822164 1.55318963   0.7006367 4.837356e-01
```

```
confint(mod.ksink, level = 0.90)
```

```
                    5 %       95 %
(Intercept) 71.152983537 92.8379375
sbp1        -0.718473464 -0.6120453
age         -0.001781907  0.1977929
bmi1        -0.070698333  0.1791684
diabetes2   -3.057148847  1.1832429
tobacco2    -3.677716789  0.9938256
tobacco3    -1.469539741  3.6459830
```

Our model is 82 - 0.67 sbp1 + 0.10 age + 0.05 bmi1 - 0.94 diabetes - 1.34 tobacconever + 1.09 tobaccoquit.

This implies that:

- for every 1 mm Hg increase in `sbp1`, we anticipate a drop in the outcome (difference in SBP) of 0.67 mm Hg (90% confidence interval: -0.73, -0.60). If we had two subjects with the same values of all other variables, but A had a baseline SBP of 150 and B had a baseline SBP of 140, then if all other variables

are kept at the same value, our model predicts that subject A's SBP will fall by 6.7 additional (90% CI: 6.0, 7.3) mm Hg as compared to subject B.

Please prepare this level of detail for at least one predictor. For the others, a summary like the one that follows will be fine.

Our kitchen sink model, within our training sample, predicts that ...

- an increase in age of 1 year is associated with a non-significant increase of 0.10 (90% CI -0.02, 0.22) mm Hg of change in SBP.
- an increase in baseline BMI of one kg/m^2 is associated with a non-significant increase of 0.05 (90% CI -0.09, 0.20) mm Hg of change in SBP.
- subjects without diabetes are associated with a non-significant decrease of 0.94 (90% CI for decrease is -1.59, 3.46) mm Hg of change in SBP as compared to subjects with diabetes.
- subjects who quit using tobacco are estimated to have a change in SBP that is 1.09 mm Hg larger than those who currently use tobacco, and subjects who never used tobacco are estimated to have a change that is 1.35 mm smaller than those who currently use. None of the differences between tobacco use groups are statistically significant at the 10% level in our training sample.

11 Step 5. Build a second model (probably with stepwise regression), and describe/assess it.

11.1 Revised Instructions

Build a second linear regression model using a subset of your four predictors, chosen by you to maximize predictive value within your training sample.

- Specify the method you used to obtain this new model. (Backwards stepwise elimination is a likely approach in many cases, but if that doesn't produce a new model, feel free to select two of your more interesting predictors from the kitchen sink model and run that as a new model.)

11.2 R code

```
step(mod.ksink)
```

```
Start:  AIC=4495.69
sbp_diff ~ sbp1 + age + bmi1 + diabetes + tobacco

           Df Sum of Sq    RSS    AIC
- bmi1      1       141 218193 4494.2
- diabetes  1       146 218198 4494.2
- tobacco   2       715 218767 4494.3
<none>                 218052 4495.7
- age       1       720 218773 4496.3
- sbp1      1    116691 334743 4836.2

Step:  AIC=4494.21
sbp_diff ~ sbp1 + age + diabetes + tobacco

           Df Sum of Sq    RSS    AIC
- tobacco   2       638 218831 4492.5
- diabetes  1       190 218383 4492.9
<none>                 218193 4494.2
```

```
- age       1       603 218796 4494.4
- sbp1      1    116815 335008 4834.8

Step:  AIC=4492.54
sbp_diff ~ sbp1 + age + diabetes

           Df Sum of Sq    RSS    AIC
- diabetes  1       142 218974 4491.1
- age       1       340 219171 4491.8
<none>                  218831 4492.5
- sbp1      1    116437 335268 4831.4

Step:  AIC=4491.06
sbp_diff ~ sbp1 + age

       Df Sum of Sq    RSS    AIC
- age   1       365 219338 4490.4
<none>              218974 4491.1
- sbp1  1    116529 335502 4830.0

Step:  AIC=4490.39
sbp_diff ~ sbp1

       Df Sum of Sq    RSS    AIC
<none>              219338 4490.4
- sbp1  1    116230 335568 4828.1


Call:
lm(formula = sbp_diff ~ sbp1, data = hbp_s_training)

Coefficients:
(Intercept)         sbp1
    88.0673      -0.6605
```

The backwards selection stepwise approach suggests a model with `sbp1` alone.

11.2.1 What if stepwise regression doesn't suggest a new model?

If stepwise regression retains the kitchen sink model, develop an alternate model by selecting a subset of the kitchen sink predictors on your own. Your kitchen sink model has at least four predictors - reduce that to the two predictors you're more interested in, and see how that model performs in what follows.

12 Step 6. Compare the two models within the training sample.

12.1 Revised Instructions

Compare this new (second) model to your "kitchen sink" model within your training sample using adjusted R^2, the residual standard error, AIC and BIC.

- Specify the complete regression equation in both models, based on the training sample.
- Which model appears better in these comparisons of the four summaries listed above? Produce a table to summarize your results. Does one model "win" each competition in the training sample?

12.2 R Code

```
mod.sbponly <- lm(sbp_diff ~ sbp1, data = hbp_s_training)
summary(mod.sbponly)
```

```

Call:
lm(formula = sbp_diff ~ sbp1, data = hbp_s_training)

Residuals:
    Min      1Q  Median      3Q     Max
-45.339 -11.301  -0.152   9.810  54.210

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 88.06727    4.42189   19.92   <2e-16 ***
sbp1        -0.66045    0.03214  -20.55   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 16.59 on 797 degrees of freedom
Multiple R-squared:  0.3464,    Adjusted R-squared:  0.3455
F-statistic: 422.3 on 1 and 797 DF,  p-value: < 2.2e-16
```

```
confint(mod.sbponly)
```

```
                2.5 %     97.5 %
(Intercept) 79.3873400 96.7472040
sbp1        -0.7235386 -0.5973701
```

The two models are specified by the coefficient estimates below.

```
pander(mod.ksink$coefficients)
```

(Intercept)	sbp1	age	bmi1	diabetes2	tobacco2	tobacco3
82	-0.6653	0.09801	0.05424	-0.937	-1.342	1.088

```
pander(mod.sbponly$coefficients)
```

(Intercept)	sbp1
88.07	-0.6605

Next, we'll compare the two models in terms of some key statistical summaries.

```
AIC(mod.ksink); AIC(mod.sbponly)
```

```
[1] 6765.158
```

```
[1] 6759.857
```

```
BIC(mod.ksink); BIC(mod.sbponly)
```

```
[1] 6802.625
```

```
[1] 6773.907
```

Model	adjusted R^2	Resid SE	AIC	BIC
Kitchen Sink	0.345	16.6	6765	6803
SBP only	0.346	16.6	6760	6774

It looks like the model with `sbp1` alone performs slightly better in the training sample, although the two models have the same residual standard error.

13 Step 7. Compare the models' predictive ability in the test sample.

13.1 Revised Instructions

Now, use your two regression models to predict the value of your outcome using the predictor values you observe in the test sample. Be sure to back-transform the predictions to the original units if you wound up fitting a model to a transformed outcome.

- Compare the two models in terms of mean squared prediction error and mean absolute prediction error in a Table, which Dr. Love will **definitely want to see** in your portfolio.
- Which model appears better at out-of-sample prediction according to these comparisons, and how do you know?

13.2 R Code

```
model.ks.predictions <- predict(mod.ksink, newdata = hbp_s_test)
model.sbponly.predictions <- predict(mod.sbponly, newdata = hbp_s_test)

model.ks.errors <- hbp_s_test$sbp_diff - model.ks.predictions
model.sbponly.errors <- hbp_s_test$sbp_diff - model.sbponly.predictions

model.ks.abserrors <- abs(model.ks.errors)
model.sbponly.abserrors <- abs(model.sbponly.errors)

model.ks.sqerrors <- model.ks.errors^2
model.sbponly.sqerrors <- model.sbponly.errors^2

summary(model.ks.abserrors)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.1658  4.9709 11.1735 13.5498 18.7922 58.2439
```

```
summary(model.ks.sqerrors)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
   0.027   24.710  124.847  307.454  353.151 3392.348
```

```
summary(model.sbponly.abserrors)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.1709  4.8088 10.9907 13.5881 18.4959 60.3963
```

```
summary(model.sbponly.sqerrors)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
   0.029   23.125  120.795  308.039  342.099 3647.718
```

Model	MAPE	MSPE	Maximum Abs. Error
Kitchen Sink	13.55	307.5	58.2
sbp1 only	13.59	308.0	60.4

So, the kitchen sink model also looks slightly better in these out-of-sample predictions.

14 Step 8. Pick a winning model, and assess regression assumptions.

14.1 Revised Instructions

Select the better of your two models (based on the results you obtain in Questions 6 and 7) and apply it to the entire data set[4].

- Do the coefficients or summaries the model show any important changes when applied to the entire data set, and not just the training set?
- Plot residuals against fitted values, and also a Normal probability plot of the residuals, each of which Dr. Love **will be looking for** in your portfolio.
- What do you conclude about the validity of standard regression assumptions for your final model based on these two plots?

14.2 R Code

I will choose the kitchen sink model. First, we apply the model to the full `hbp_s` data set.

```
model.final <- lm(sbp_diff ~ sbp1 + age + bmi1 + diabetes + tobacco, data = hbp_s)
summary(model.final)
```

```
Call:
lm(formula = sbp_diff ~ sbp1 + age + bmi1 + diabetes + tobacco,
    data = hbp_s)

Residuals:
   Min     1Q Median     3Q    Max
-47.76 -11.36  -0.57  10.25  58.99

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 80.92486    5.98876  13.513   <2e-16 ***
sbp1        -0.65732    0.02906 -22.621   <2e-16 ***
age          0.09693    0.05473   1.771   0.0768 .
bmi1         0.04096    0.06784   0.604   0.5461
```

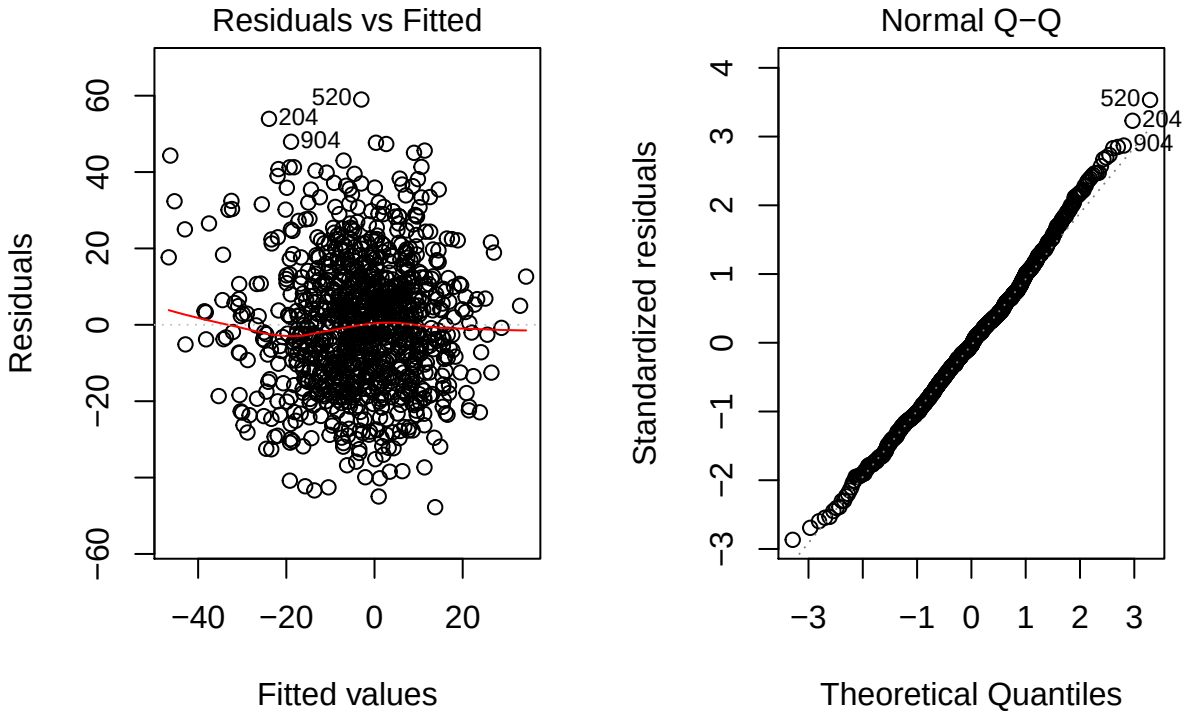
[4] If, as in my case, you have to choose between the in-sample and out-of-sample results, I would likely select the out-of-sample results to choose my final model.

```
diabetes2   -0.43342    1.14114  -0.380   0.7042
tobacco2    -2.14346    1.28122  -1.673   0.0946 .
tobacco3     0.19734    1.39290   0.142   0.8874
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 16.76 on 992 degrees of freedom
Multiple R-squared:  0.3416,    Adjusted R-squared:  0.3376
F-statistic: 85.79 on 6 and 992 DF,  p-value: < 2.2e-16
```

At the 90% confidence level, it appears that age and (part of) tobacco usage now appear to be statistically significant in our t tests. The overall R^2 is very comparable, as is the residual standard error, to the model fit to the training sample alone. No coefficients change their signs.

Here are the residual plots.

```
par(mfrow = c(1,2))
plot(model.final, which = 1:2)
```




```
par(mfrow = c(1,1))
```

I see no substantial violations of regression assumptions. There is neither a curve, nor a fan shape in the residuals vs. fitted values, and we see no evidence of important non-Normality in the Normal Q-Q plot.