University of Toronto Scarborough
Department of Computer and Mathematical Sciences
STAC32 (K. Butler), Final Exam
December 9, 2022

Aids allowed (on paper, no computers):

- My lecture overheads (slides)
- Any notes that you have taken in this course
- Your marked assignments
- My assignment solutions
- Non-programmable, non-communicating calculator

This exam has 9 numbered pages of questions plus this cover page.
In addition, you have an additional booklet of Figures to refer to during the exam.
The maximum marks available for each part of each question are shown next to the question part.
If you need more space, use the last page of the exam. Anything written on the back of the page will not be graded.
You may assume throughout this exam that the code shown in Figure 1 of the booklet of Figures has already been run.
The University of Toronto's Code of Behaviour on Academic Matters applies to all University of Toronto Scarborough students. The Code prohibits all forms of academic dishonesty including, but not limited to, cheating, plagiarism, and the use of unauthorized aids. Students violating the Code may be subject to penalties up to and including suspension or expulsion from the University.

1. The crab species *Leptograpsus variegatus* has two colour forms, blue and orange. Fifty crabs of each colour form and both sexes were collected at Fremantle, Western Australia, and various measurements were taken on each crab. The variables in the data file are as follows:

   - `sp` species (B is blue and O is orange, a letter O)
   - `sex` (male, M or female, F)
   - `index` serial number of each crab within species and sex
   - `FL` frontal lobe size
   - `RW` rear width
   - `CL` carapace length
   - `CW` carapace width
   - `BD` body depth

   All five of the measurements are in millimetres.

   Some of the data file is shown in Figure 2, and the data file is stored in `crabs.txt` in the same folder that R Studio is running in.

   (a) [2] What code would read the data in the file into a dataframe called `crabs` and display it?

   (b) [3] What code would obtain the number of observations in each group, and the mean and SD of carapace length, for each combination of species and sex?

   (c) [4] What code would make a suitable graph of rear width for each species and sex? What is the name of this type of graph?

   (d) [3] What code would produce histograms of body depth for the two species separately, laid out above and below each other so that you can compare the distributions?

(e) [2] How many crabs are there with carapace length greater than 40 for each species? What code would determine this?

(f) [3] Out of the crabs with carapace length greater than 40, what is the smallest rear width, categorized by species? What code would calculate this?

2. Hemophilia is a bleeding disorder in which the blood does not clot properly. This can lead to bleeding that does not stop. Blood contains many proteins called clotting factors that can help to stop bleeding.

   One type of hemophilia is called Hemophilia A. In a study, 75 women were classified according to whether they do not have hemophilia A (30 women; "normal") or whether they do have hemophilia A (45 women; "carrier"). This is in the column called `gr`. Two variables were measured, called `AHFactivity` and `AHFantigen`, and the research question was whether either of them have any diagnostic value (meaning, whether either of them in any way distinguish women with hemophilia A from women without hemophilia A.)

   Some of the data are shown in Figure 3.

   (a) [2] Describe *in words* a graph that would help to determine whether either of the measured variables help to distinguish women with hemophilia A from women without.

   (b) [2] Explain briefly how you would use your graph of the previous part to determine whether either of the two measured variables distinguish women with hemophilia A from women without hemophilia A. If you find it helpful, use an example or a sketch of the kind of graph you might get to support your explanation.

(c) [3] The researchers decided to use `AHFactivity` to compare the two groups of women. A graph is shown in Figure 4. Two possible analyses are shown in Figures 5 and 6. Which analysis do you prefer and why? Explain briefly.

(d) [2] What do you conclude from your preferred test, in the context of the data?

(e) [2] Figure 7 shows bootstrap sampling distributions of the sample mean for each group of women. Do these plots support your conclusions about which test to do, or do they cause you to change your mind? Explain briefly.

(f) [2] Why did we not do a one-sided test, despite the evidence in Figure 4?

3. This question is all about tidying data: that is, rearranging it to display the data in a different layout that might be more convenient for graphing, display, etc.

   (a) [3] Figure 8 shows a dataframe `d1` that needs to be rearranged as the dataframe `d2` shown in Figure 9. What code would rearrange `d1` into `d2`?

   (b) [2] Suppose, instead, you had been given the dataframe `d2` shown in Figure 9 and were asked to rearrange it into `d1`. What code would do this?

   (c) [4] A dataframe `dd` is shown in Figure 10. The numerical values are all of a variable `y` observed under different conditions: a `level` that takes the values `Hi` and `Lo`, and a `size` that takes the values `Large` and `Small`. There are two replicate values of `y` observed at each level-size combination; these are labelled `R1` and `R2` in the column `rep`.

   It is desired to arrange the dataframe so that all the values of `y` are in one column, with columns indicating the `level`, `size`, and `rep` at which that value of `y` was observed. What is the most concise code that would do this?

   (d) [4] A dataframe `ddd` is shown in Figure 11. Some code is shown in Figure 12. What output will that code produce?

4. 164 men took part in an experiment to see whether the drug cholostyramine lowered blood cholesterol levels. The men were supposed to take six packets of cholostyramine per day, but many actually took much less. We want to investigate whether men that better adhered to the instructions had a greater improvement in blood cholesterol levels.

   Some of the data are shown in Figure 13. There are two columns:

   - `compliance`: the percentage of all the packets of cholostyramine given to that man that were actually taken
   - `improvement`: the blood cholesterol was measured at the beginning and at the end of the study. This column is the percentage decrease, compared to the initial value. A negative `improvement` means that blood cholesterol was worse (higher) at the end of the study than it was at the beginning.

   (a) [4] A scatterplot is shown in Figure 14. Interpret this scatterplot in terms of the form, direction, and strength of the relationship. ("Form" means whether it is linear or curved or something else, "direction" is up or down, "strength" is strong, weak, or something in between.) Explain briefly.

   (b) [2] The output from a linear regression is shown in Figure 15. Do you think there is there a real relationship between compliance and improvement, on the basis of this output? Explain briefly.

   (c) [2] A plot of residuals from the regression `cholost.1` is shown in Figure 16. What do you conclude from this plot? Explain briefly.

   (d) [2] Another plot of residuals is shown in Figure 17. What do you conclude from this plot? Explain briefly.

(e) [3] One of the researchers believes that a compliance over 95 (percent) is associated with better improvement, and fitted the model whose code and output are shown in Figure 18. Is there a significant benefit to a compliance over 95 percent, over and above the greater improvement that we have already seen goes with a greater compliance? How big is this benefit? How can you tell? (Note that in a regression model, a true-false variable is treated the same as a categorical variable with levels `TRUE` and `FALSE`).

5. Patients who require the same treatment can be charged different amounts of money, even by the same hospital. Are there systematic reasons why this is the case? A doctor collected data on 49 patients with the same diagnosis at the same (large) hospital, as follows:

   - `Sex` M (Male) or F (Female).
   - `MD` which doctor they were treated by (there are three different doctors)
   - `Svty` severity of illness, from 1 (lowest severity) to 4 (highest), which we treat as quantitative.
   - `Chrg` total amount charged by the hospital, in dollars (response)
   - `Age` of patient in years

   Some of the data are shown in Figure 19.

   (a) [2] Plots of charges against each of the explanatory variables are shown in Figures 20 and 21. Why are some of the plots boxplots?

   (b) [2] There was one patient whose charge was much higher than for any of the other patients. What do Figures 20 and 21 tell you about that patient?

   (c) [2] Ignoring the upper outlier, what do Figures 20 and 21 tell you about when charges are higher, for each of the four explanatory variables? (Four very short answers.)

(d) [2] A regression model is fitted, with output shown in Figure 22. Assuming that the residual plots look appropriate, what would you do next, and why is the `drop1` output better to decide this from?

(e) [2] In Figure 22, do the positive Estimates for severity and age make practical sense, in the context of the data? Explain briefly.

(f) [2] Based on Figure 22, which doctor has the highest predicted charges, all else equal? Explain briefly.

(g) [2] What is the precise meaning of the P-value 0.0368 in Figure 22?

6. The function rnorm generates random normal data. It has three inputs: the sample size, the population mean, and the population standard deviation. Let's suppose that you will be generating a lot of normal random numbers with various sample sizes and various means, but the population standard deviation will always be half as big as the mean. You want to streamline your process by writing a function called my_random that has as input a sample size n and a population mean mu, and generates and returns a normal random sample according to the specifications.

   (a) [3] What code would you use to write your function?

   (b) [2] How would you use your function to obtain 7 random normal values with mean 10 and SD 5?

   (c) [3] For the next little while, you are told that the sample size will be 10. How would you *change* your function to avoid having to enter the sample size if it is 10, and how could you most concisely use your new function to obtain 10 random normal numbers with mean 20 (and SD 10)? (You only need to give what *changes* you would make.)

   (d) [3] Figure 23 shows a dataframe d containing some population means. How would you use your modified function to make a list-column called sample_data containing random samples of size 10 from normal distributions with the appropriate means (and standard deviations that are half as big as the means), without using a loop? Give the code you would use.

   (e) [2] How would you arrange it so that you could see the actual random data that had been generated? (Give the *addition* to your previous code.)

Use the rest of this page if you need more space. Be sure to label any answers here with the question and part they belong to.

Figures

```
library(tidyverse)
library(readxl)
library(smmr)
```

Figure 1: Packages

```
sp:sex:index:FL:RW:CL:CW:BD
B:M:1:8.1:6.7:16.1:19:7
B:M:2:8.8:7.7:18.1:20.8:7.4
B:M:3:9.2:7.8:19:22.4:7.7
B:M:4:9.6:7.9:20.1:23.1:8.2
B:F:3:9.1:8.1:18.5:21.6:7.7
B:F:4:9.1:8.2:19.2:22.2:7.7
B:F:5:9.5:8.2:19.6:22.4:7.8
B:F:6:9.8:8.9:20.4:23.9:8.8
O:M:10:13.7:11:27.5:30.5:12.2
O:M:11:14:11.5:29.2:32.2:13.1
O:M:12:14.1:10.4:28.9:31.8:13.5
O:M:13:14.1:10.5:29.1:31.6:13.1
O:F:36:19.7:16.7:39.9:43.6:18.2
O:F:37:19.9:16.6:39.4:43.9:17.9
O:F:38:19.9:17.9:40.1:46.4:17.9
O:F:39:20:16.7:40.4:45.1:17.7
```

Figure 2: Crabs data (some)

```
hemophilia %>% slice_sample(n = 20)
```

```
##    AHFactivity AHFantigen      gr
## 22      0.1507     0.0933  normal
## 38     -0.4535    -0.1682 carrier
## 44     -0.4319    -0.0687 carrier
## 29     -0.1972    -0.0607  normal
## 49     -0.5107    -0.2483 carrier
## 9      -0.1913    -0.2123  normal
## 35     -0.1326     0.0097 carrier
## 20     -0.2015    -0.0498  normal
## 45     -0.2734    -0.0020 carrier
## 67     -0.0964     0.0531 carrier
## 14      0.0084     0.0782  normal
## 72     -0.1744     0.1892 carrier
## 1      -0.0056    -0.1657  normal
## 43     -0.3226     0.1670 carrier
## 46     -0.5573     0.0548 carrier
## 36     -0.6911    -0.3390 carrier
## 19      0.0006    -0.1153  normal
## 51     -0.2447    -0.0407 carrier
## 75     -0.4784     0.0282 carrier
## 74     -0.2444     0.1614 carrier
```

Figure 3: Hemophilia data (20 randomly chosen rows)

```
ggplot(hemophilia, aes(x = gr, y = AHFactivity)) + geom_boxplot()
```

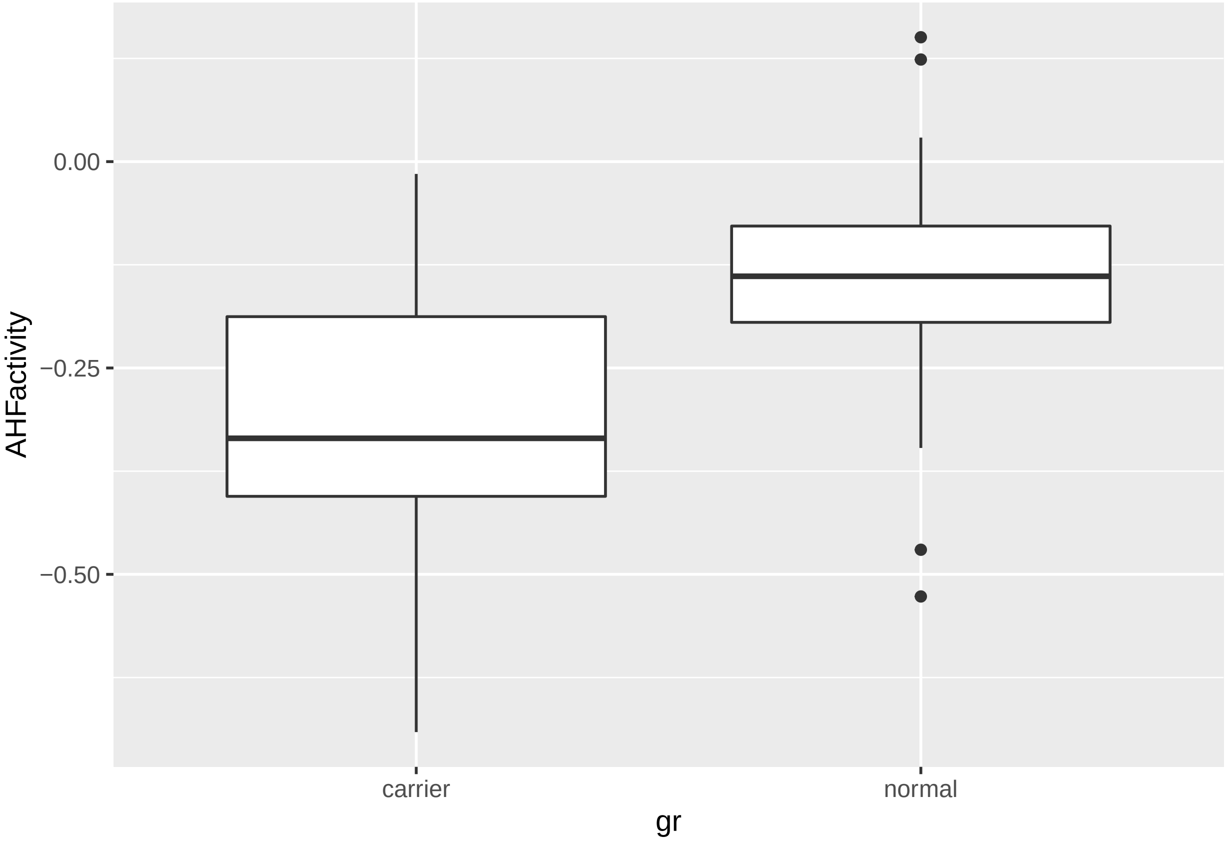

Figure 4: Graph of `AHFactivity` for each group of women

```
t.test(AHFactivity ~ gr, data = hemophilia)
```

```
##
##  Welch Two Sample t-test
##
## data:  AHFactivity by gr
## t = -4.9448, df = 65.029, p-value = 5.655e-06
## alternative hypothesis: true difference in means between group carrier and group normal is not equal
## 95 percent confidence interval:
##  -0.2429789 -0.1031744
## sample estimates:
## mean in group carrier  mean in group normal
##            -0.3079467            -0.1348700
```

Figure 5: Test 1 for hemophilia data

```
median_test(hemophilia, AHFactivity, gr)
```

```
## $table
##          above
## group     above below
##   carrier    12    33
##   normal     25     4
##
## $test
##        what        value
## 1 statistic 2.500690e+01
## 2        df 1.000000e+00
## 3   P-value 5.712562e-07
```

Figure 6: Test 2 for hemophilia data

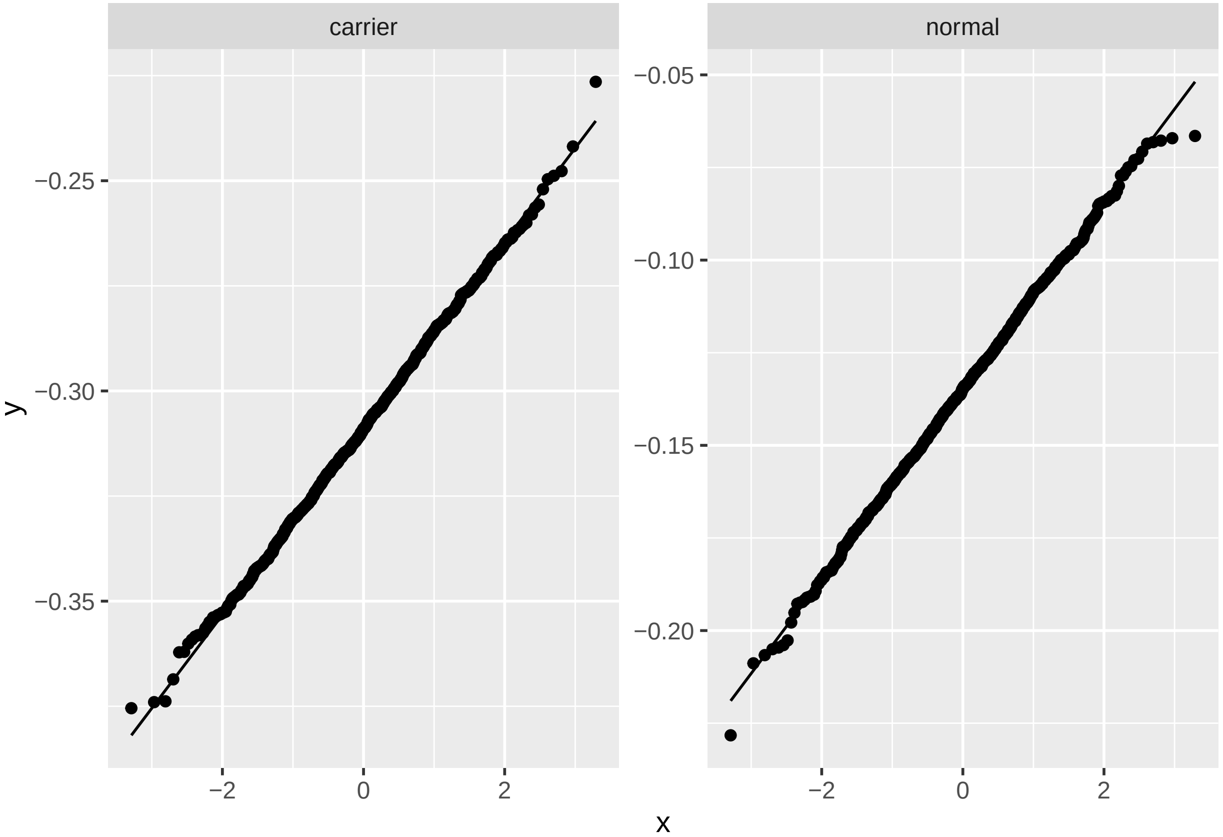

Figure 7: Bootstrap sampling distributions of sample means for hemophilia data, normal quantile plots

```
d1
```

```
## # A tibble: 3 x 4
##   id       g1    g2    g3
##   <chr> <dbl> <dbl> <dbl>
## 1 A        10    21    29
## 2 B        11    20    28
## 3 C        12    22    31
```

Figure 8: Dataframe `d1`

```
d2
```

```
## # A tibble: 9 x 3
##   id    treatment score
##   <chr> <chr>     <dbl>
## 1 A     g1           10
## 2 A     g2           21
## 3 A     g3           29
## 4 B     g1           11
## 5 B     g2           20
## 6 B     g3           28
## 7 C     g1           12
## 8 C     g2           22
## 9 C     g3           31
```

Figure 9: Dataframe `d2`

```
dd
```

```
## # A tibble: 2 x 5
##   rep   HiLarge HiSmall LoLarge LoSmall
##   <chr>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 R1         16      17      19      18
## 2 R2         18      20      22      21
```

Figure 10: Dataframe `dd`

```
ddd
```

```
## # A tibble: 4 x 3
##   id    g         y
##   <chr> <chr> <dbl>
## 1 A     lo       20
## 2 B     hi       22
## 3 C     lo       23
## 4 D     hi       24
```

Figure 11: Dataframe `ddd`

```
ddd %>%
  pivot_wider(names_from = id, values_from = y)
```

Figure 12: Code to run on dataframe `ddd`

```
cholost %>% slice(1:20)
```

```
##    compliance improvement
## 1           0       -5.25
## 2          27       -1.50
## 3          71       59.50
## 4          95       32.50
## 5           0       -7.25
## 6          28       23.50
## 7          71       14.75
## 8          95       70.75
## 9           0       -6.25
## 10         29       33.00
## 11         72       63.00
## 12         95       18.25
## 13          0       11.50
## 14         31        4.25
## 15         72        0.00
## 16         95       76.00
## 17          2       21.00
## 18         32       18.75
## 19         73       42.00
## 20         95       75.75
```

Figure 13: Cholostyramine data (some)

```
ggplot(cholost, aes(x = compliance, y = improvement)) + geom_point()
```

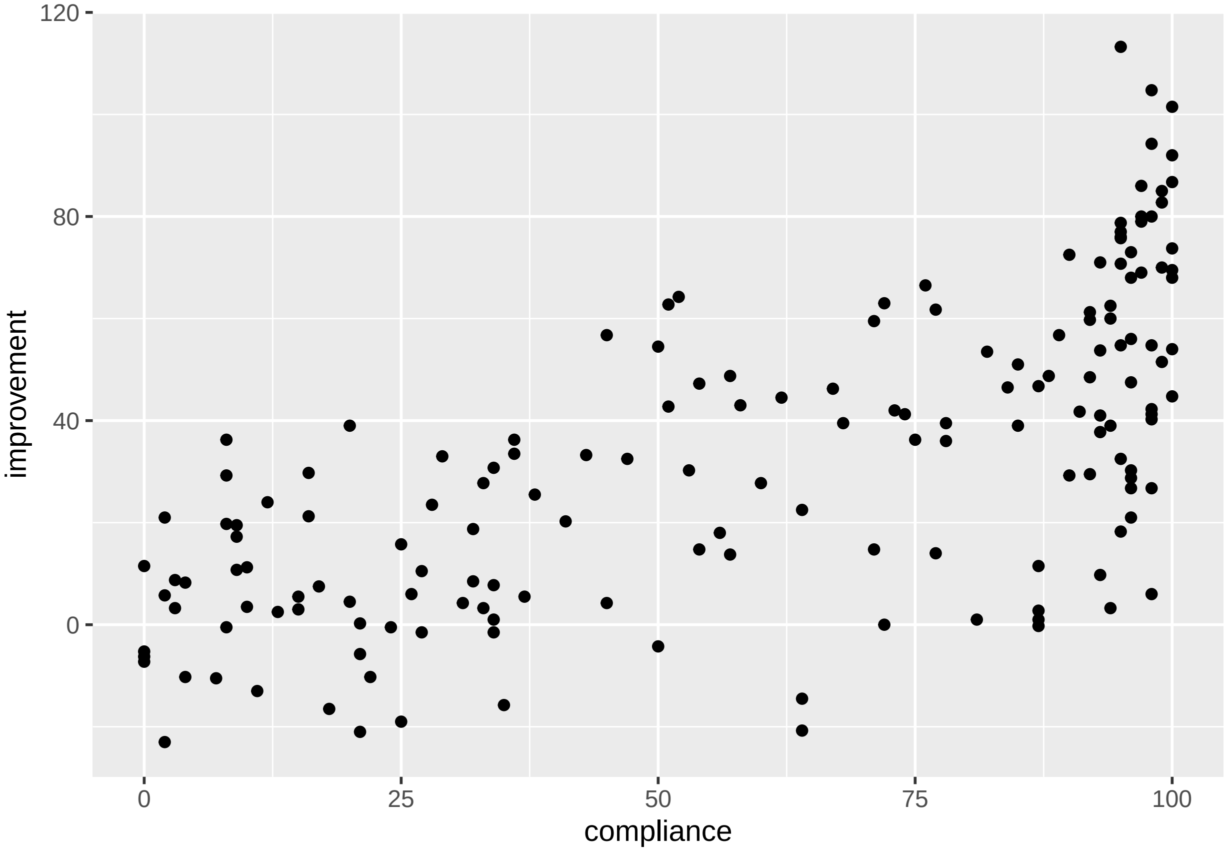

Figure 14: Cholostyramine scatterplot

```
cholost.1 <- lm(improvement ~ compliance, data = cholost)
summary(cholost.1)
```

```
##
## Call:
## lm(formula = improvement ~ compliance, data = cholost)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -55.83 -13.69   0.15  15.59  60.07
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.30725    3.44903  -0.669    0.504
## compliance   0.58410    0.04967  11.760   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.11 on 162 degrees of freedom
## Multiple R-squared:  0.4605, Adjusted R-squared:  0.4572
## F-statistic: 138.3 on 1 and 162 DF,  p-value: < 2.2e-16
```

Figure 15: Cholostyramine regression 1

```
ggplot(cholost.1, aes(sample = .resid)) + stat_qq() + stat_qq_line()
```

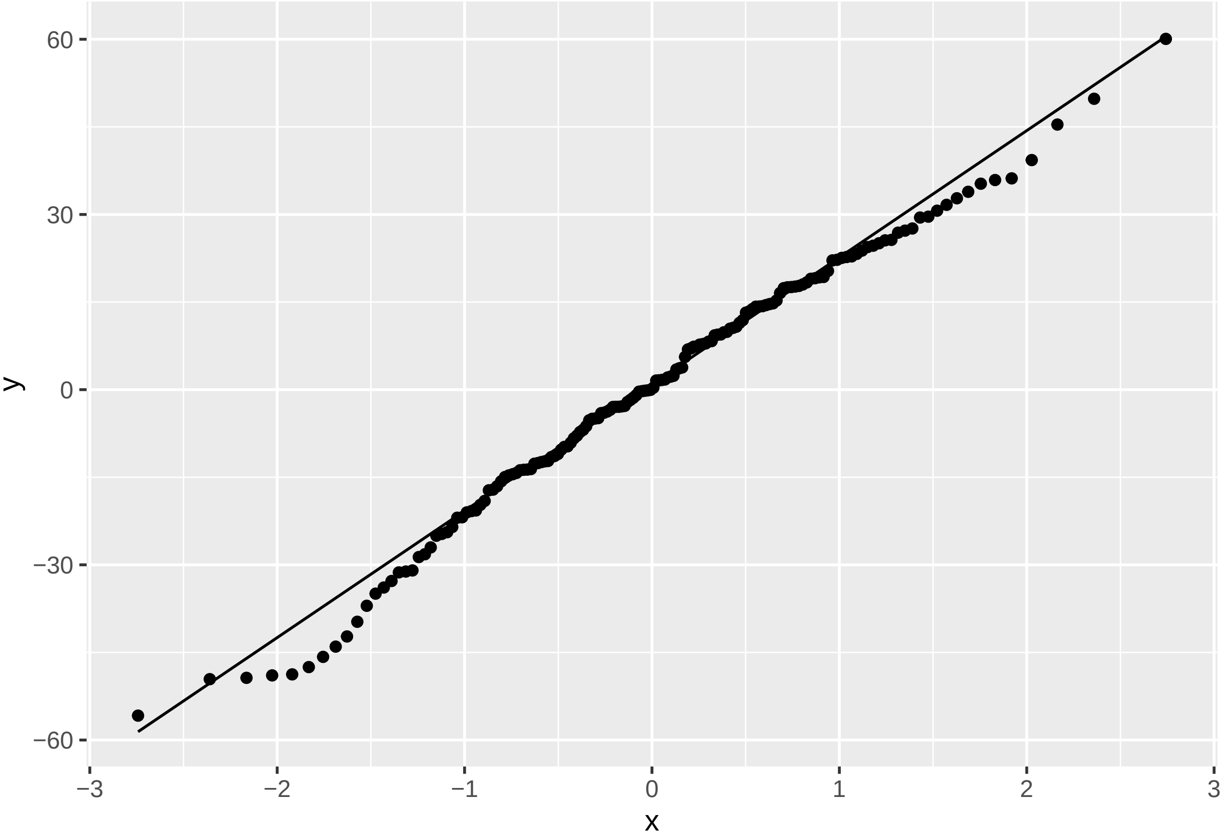

Figure 16: Residual plot 1 for cholostyramine data

```
ggplot(cholost.1, aes(x = .fitted, y = .resid)) + geom_point()
```

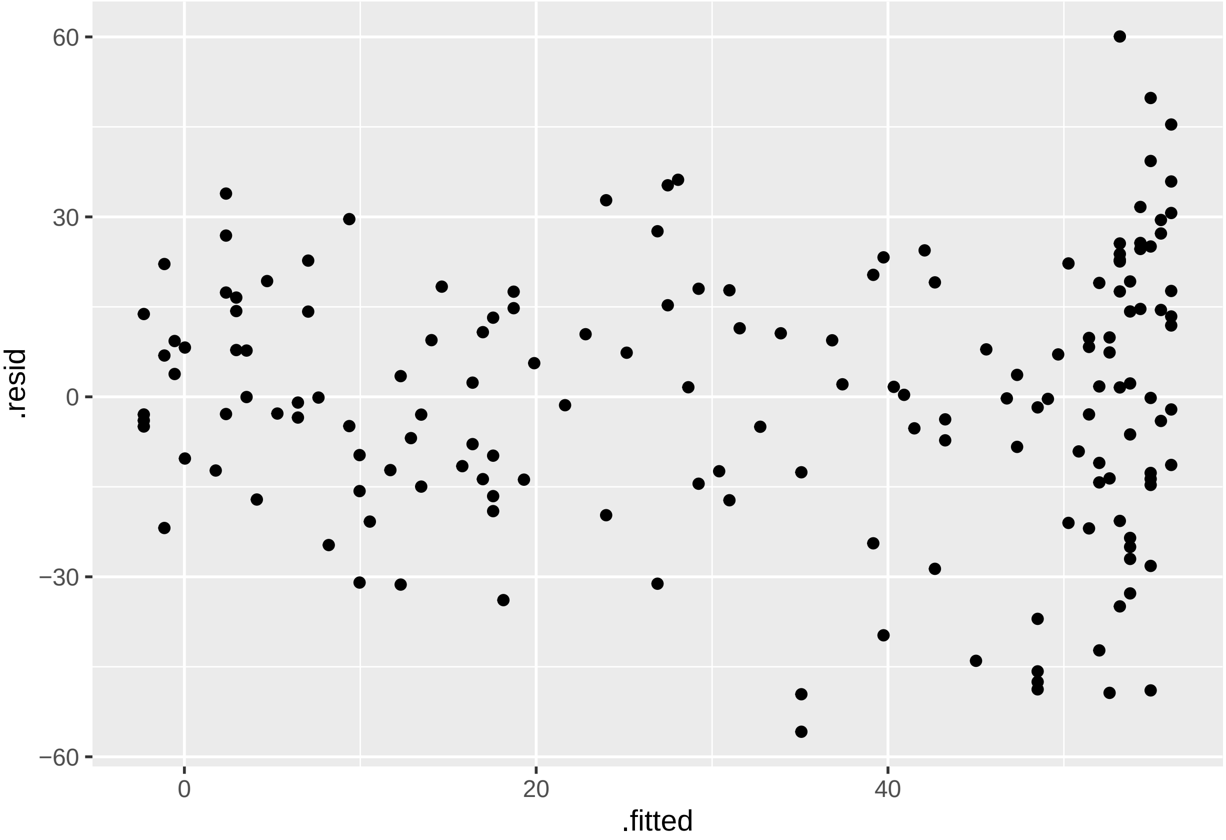

Figure 17: Residual plot 2 for cholostyramine data

```
cholost %>% mutate(bonus = (compliance >= 95)) -> cholost_bonus
cholost.2 <- lm(improvement ~ compliance + bonus, data = cholost_bonus)
summary(cholost.2)
```

```
##
## Call:
## lm(formula = improvement ~ compliance + bonus, data = cholost_bonus)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.951 -12.987   3.153  15.667  51.625
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.63250    3.49127   0.468  0.64071
## compliance   0.44178    0.06154   7.179 2.45e-11 ***
## bonusTRUE   18.02349    4.89995   3.678  0.00032 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.3 on 161 degrees of freedom
## Multiple R-squared:  0.5023, Adjusted R-squared:  0.4962
## F-statistic: 81.26 on 2 and 161 DF,  p-value: < 2.2e-16
```

Figure 18: Another regression for the cholostyrine data

```
charges %>% slice_sample(n = 20)
```

```
##     Sex     MD Svty  Chrg Age
## 3     M  MD730    1  1487  17
## 30    F  MD499    1  2499  39
## 28    M  MD499    3 15600  72
## 24    M  MD499    2  3535  20
## 12    F  MD730    2 14111  85
## 6     M  MD730    3 20280  61
## 18    F  MD730    3 24809  73
## 8     M  MD730    3 22382  90
## 37    F MD1021    4 64465  71
## 11    F  MD730    4 22642  77
## 33    M  MD499    3 15969  60
## 27    F  MD499    3 24121  86
## 5     M  MD730    2 18823  61
## 44    M MD1021    2  8759  56
## 38    F MD1021    3 17506  71
## 14    F  MD730    2 13343  65
## 7     F  MD730    1  4360  44
## 43    F MD1021    3 22734  66
## 31    M  MD499    3 12423  69
## 1     M  MD730    2  8254  57
```

Figure 19: Hospital charges data (20 randomly chosen rows)

```
charges %>%
  pivot_longer(c(Svty, Age)) %>%
  ggplot(aes(x = value, y = Chrg)) + geom_point() +
  facet_wrap(~name, scales = "free")
```

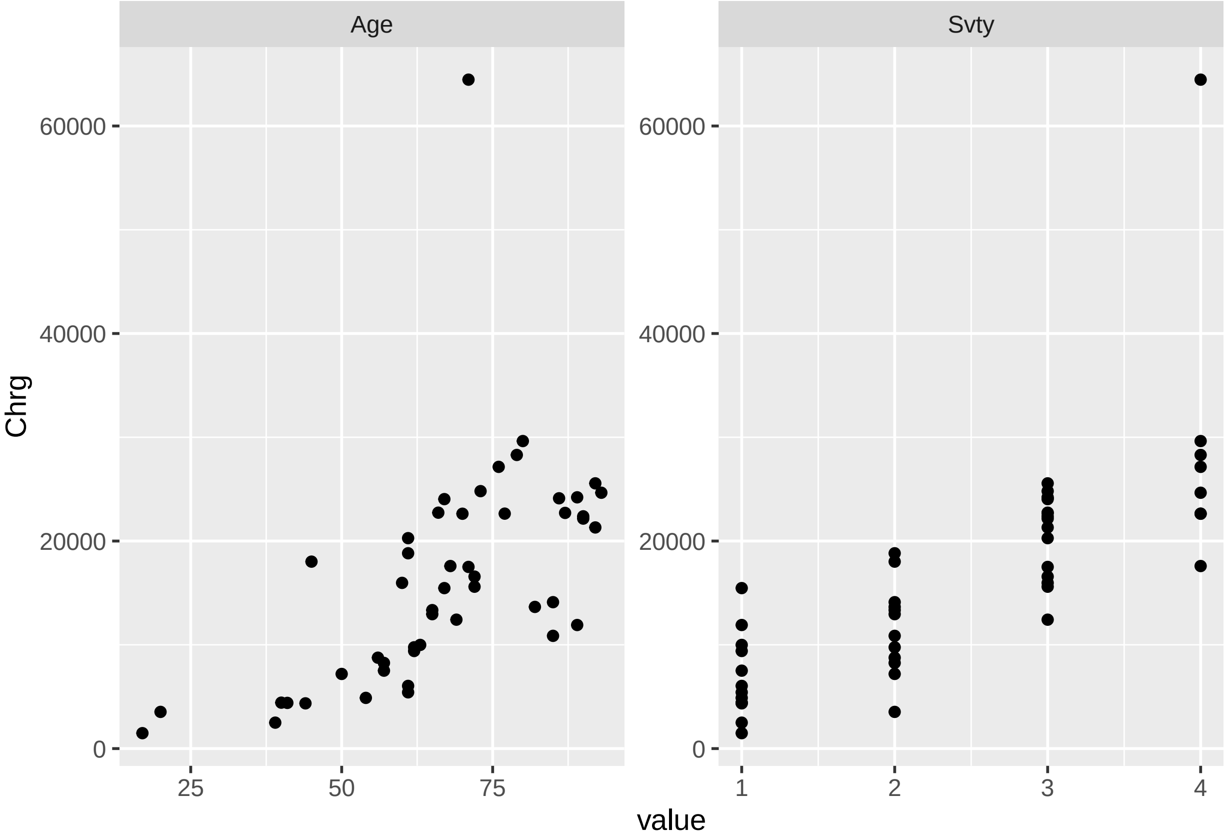

Figure 20: Plot of hospital charges against explanatory variables part 1

```
charges %>%
  pivot_longer(c(Sex, MD)) %>%
  ggplot(aes(x = value, y = Chrg)) + geom_boxplot() +
    facet_wrap(~name, scales = "free")
```

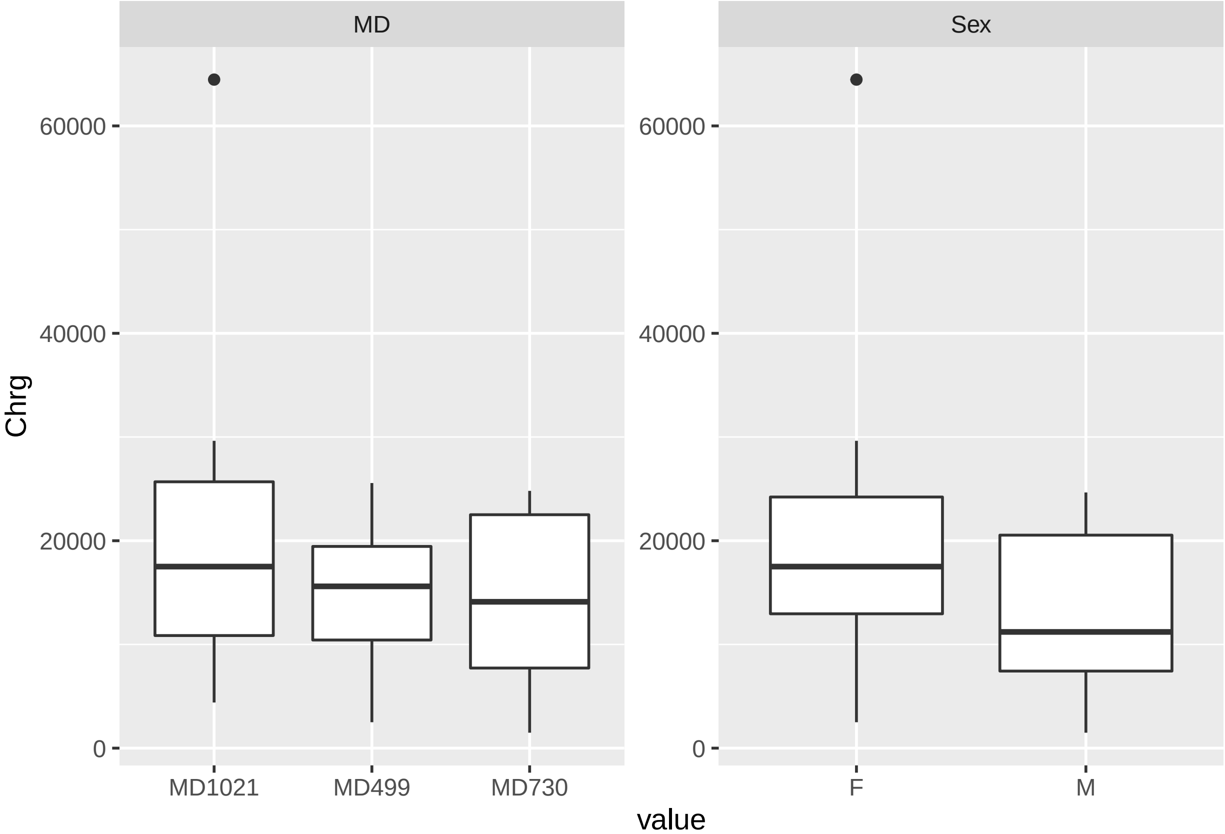

Figure 21: Plot of hospital charges against explanatory variables part 2

```
charges.1 <- lm(Chrg ~ Sex + MD + Svty + Age, data = charges)
summary(charges.1)
```

```
##
## Call:
## lm(formula = Chrg ~ Sex + MD + Svty + Age, data = charges)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -7435  -3094   -924   1661  33883
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3556.67    4211.82  -0.844   0.4031
## SexM        -1178.13    2076.91  -0.567   0.5735
## MDMD499     -5176.48    2402.16  -2.155   0.0368 *
## MDMD730     -3878.69    2389.86  -1.623   0.1119
## Svty         6292.14    1054.71   5.966  4.1e-07 ***
## Age           126.34      65.95   1.916   0.0621 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6405 on 43 degrees of freedom
## Multiple R-squared:  0.6684, Adjusted R-squared:  0.6299
## F-statistic: 17.34 on 5 and 43 DF,  p-value: 2.273e-09
```

```
drop1(charges.1, test = "F")
```

```
## Single term deletions
##
## Model:
## Chrg ~ Sex + MD + Svty + Age
##        Df  Sum of Sq        RSS    AIC F value    Pr(>F)
## <none>               1763818288 864.55
## Sex     1   13198805 1777017093 862.91  0.3218   0.57349
## MD      2  201004856 1964823144 865.84  2.4501   0.09824 .
## Svty    1 1459873008 3223691295 892.10 35.5901 4.101e-07 ***
## Age     1  150508850 1914327138 866.56  3.6692   0.06209 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 22: Regression model and output

```
d
```

```
## # A tibble: 3 x 1
##   the_mean
##      <dbl>
## 1        4
## 2        8
## 3       24
```

Figure 23: Population means to use with your function for generating random normal data