Figures

```
library(MASS)
library(tidyverse)
library(smmr)
library(cmdstanr)
```

Figure 1: Packages

```
cultivar_name,alcohol,malic_acid,ash,mg
grignolino,12.87,4.61,2.48,86
grignolino,13.4,4.6,2.86,112
barolo,13.9,1.68,2.12,101
barbera,11.84,0.89,2.58,94
barolo,13.51,1.8,2.65,110
barolo,13.05,1.73,2.04,92
barolo,14.38,1.87,2.38,102
barolo,13.5,1.81,2.61,96
barbera,11.87,4.31,2.39,82
barbera,12,0.92,2,86
barbera,12.21,1.19,1.75,151
barbera,12.52,2.43,2.17,88
```

Figure 2: Wine data (some)

```
wine
```

```
## # A tibble: 178 x 5
##    cultivar_name alcohol malic_acid   ash    mg
##    <chr>           <dbl>      <dbl> <dbl> <dbl>
##  1 grignolino       12.9       4.61  2.48    86
##  2 grignolino       13.4       4.6   2.86   112
##  3 barolo           13.9       1.68  2.12   101
##  4 barbera          11.8       0.89  2.58    94
##  5 barolo           13.5       1.8   2.65   110
##  6 barolo           13.0       1.73  2.04    92
##  7 barolo           14.4       1.87  2.38   102
##  8 barolo           13.5       1.81  2.61    96
##  9 barbera          11.9       4.31  2.39    82
## 10 barbera          12         0.92  2        86
## # ... with 168 more rows
```

Figure 3: Wine data after being read in (some)

```
##
##      One-sample t test power calculation
##
##               n = 25.38969
##           delta = 10
##              sd = 20
##       sig.level = 0.05
##           power = 0.677
##     alternative = two.sided
```

Figure 4: Power analysis

```r
wine %>% group_by(cultivar_name) %>%
  summarize(n = n())
```

```
## # A tibble: 3 x 2
##   cultivar_name     n
##   <chr>         <int>
## 1 barbera          71
## 2 barolo           59
## 3 grignolino       48
```

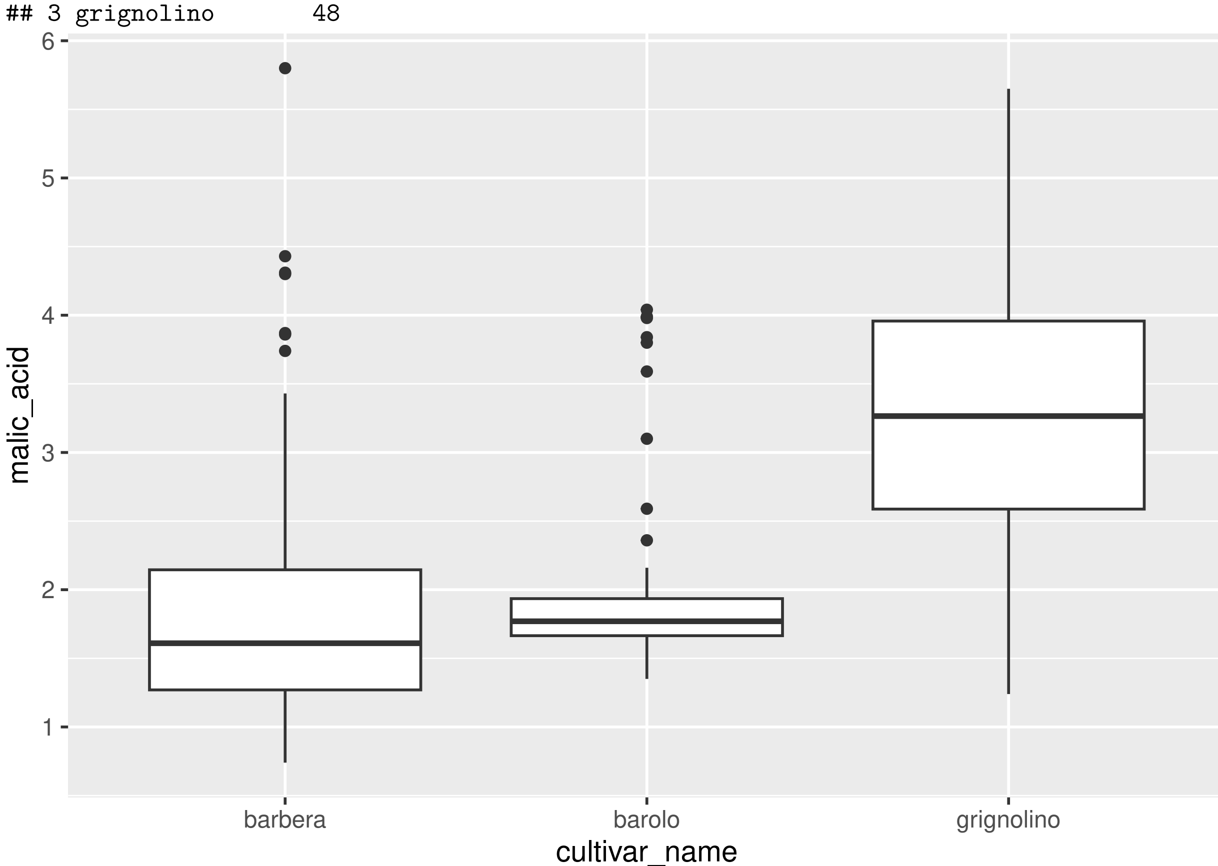

Figure 5: Wine data summary and plot

```
median_test(wine, malic_acid, cultivar_name)
```

```
## $table
##              above
## group         above below
##   barbera       25    46
##   barolo        21    38
##   grignolino    43     5
##
## $test
##        what        value
## 1 statistic 4.119291e+01
## 2        df 2.000000e+00
## 3   P-value 1.135205e-09
```

Figure 6: Wine data Mood Median Test

```
pairwise_median_test(wine, malic_acid, cultivar_name)
```

```
## # A tibble: 3 x 4
##   g1      g2          p_value adj_p_value
##   <chr>   <chr>         <dbl>       <dbl>
## 1 barbera barolo     3.97e- 2    1.19e- 1
## 2 barbera grignolino 1.52e-11    4.55e-11
## 3 barolo  grignolino 2.15e-12    6.44e-12
```

Figure 7: Wine data pairwise median tests

```
wine %>% filter(cultivar_name == "barolo") -> barolo
tibble(sim = 1:10000) %>%
  rowwise() %>%
  mutate(my_sample = list(sample(barolo$malic_acid, replace = TRUE))) %>%
  mutate(my_mean = mean(my_sample)) %>%
  ggplot(aes(sample = my_mean)) + stat_qq() + stat_qq_line()
```

Figure 8: Wine data mystery code

```
d1
```

```
## # A tibble: 2 x 3
##      id     a     b
##   <dbl> <dbl> <dbl>
## 1     1    10    11
## 2     2     8     9
```

Figure 9: Dataframe `d1`

```
d1 %>% pivot_longer(-id, names_to = "name", values_to = "value")
```

Figure 10: Code for dataframe `d1`

```
d2
```

```
## # A tibble: 2 x 5
##     row  m_ht  f_ht  m_wt  f_wt
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     7   180   150    80    60
## 2     8   185   160    90    55
```

Figure 11: Dataframe d2

```
## # A tibble: 8 x 4
##     row gender what  measure
##   <dbl> <chr>  <chr>   <dbl>
## 1     7 m      ht        180
## 2     7 f      ht        150
## 3     7 m      wt         80
## 4     7 f      wt         60
## 5     8 m      ht        185
## 6     8 f      ht        160
## 7     8 m      wt         90
## 8     8 f      wt         55
```

Figure 12: Dataframe d2 output

```
d3
```

```
## # A tibble: 2 x 3
##      id   x_1   x_2
##   <dbl> <dbl> <dbl>
## 1     4    10    11
## 2     6     8     9
```

Figure 13: Dataframe d3

```
d3 %>% pivot_longer(-id, names_to = c(".value", "col"), names_sep = "_")
```

Figure 14: Code for dataframe d3

```
d4
```

```
## # A tibble: 4 x 3
##     row x     measure
##   <dbl> <chr>   <dbl>
## 1     7 m_ht      180
## 2     7 f_ht      150
## 3     7 m_wt       80
## 4     7 f_wt       60
```

Figure 15: Dataframe d4

```
d4 %>% separate(x, into = c("gender", "what"), sep = "_")
```

Figure 16: Code for dataframe `d4`

```
d5
```

```
## # A tibble: 4 x 3
##     row group     x
##   <dbl> <chr> <dbl>
## 1     1 a        14
## 2     1 b        15
## 3     2 a        16
## 4     2 b        17
```

Figure 17: Dataframe `d5`

```
## # A tibble: 2 x 3
##     row     a     b
##   <dbl> <dbl> <dbl>
## 1     1    14    15
## 2     2    16    17
```

Figure 18: Dataframe `d5` output

```
d6
```

```
## # A tibble: 4 x 3
##   x         y z
##   <chr> <dbl> <chr>
## 1 c        16 low
## 2 b        18 high
## 3 a        20 medium
## 4 b        22 low
```

Figure 19: Dataframe `d6`

```
d6 %>% pivot_wider(names_from = z, values_from = y)
```

Figure 20: Code for dataframe `d6`

```
engel
```

```
## # A tibble: 234 x 2
##    income foodexp
##     <dbl>   <dbl>
##  1   420.    256.
##  2   541.    311.
##  3   901.    486.
##  4   639.    403.
##  5   751.    496.
##  6   946.    634.
##  7   829.    631.
##  8   979.    700.
##  9  1310.    831.
## 10  1492.    815.
## # ... with 224 more rows
```

Figure 21: Food expenditure data (some)

```
engel.1 <- lm(foodexp ~ income, data  = engel)
summary(engel.1)
```

```
##
## Call:
## lm(formula = foodexp ~ income, data = engel)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -622.00  -54.02    3.22   52.87  398.72
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 91.33302   15.52094   5.885 1.39e-08 ***
## income       0.54654    0.01458  37.497  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 100.2 on 232 degrees of freedom
## Multiple R-squared:  0.8584, Adjusted R-squared:  0.8578
## F-statistic:  1406 on 1 and 232 DF,  p-value: < 2.2e-16
```

Figure 22: Food expenditure: regression analysis

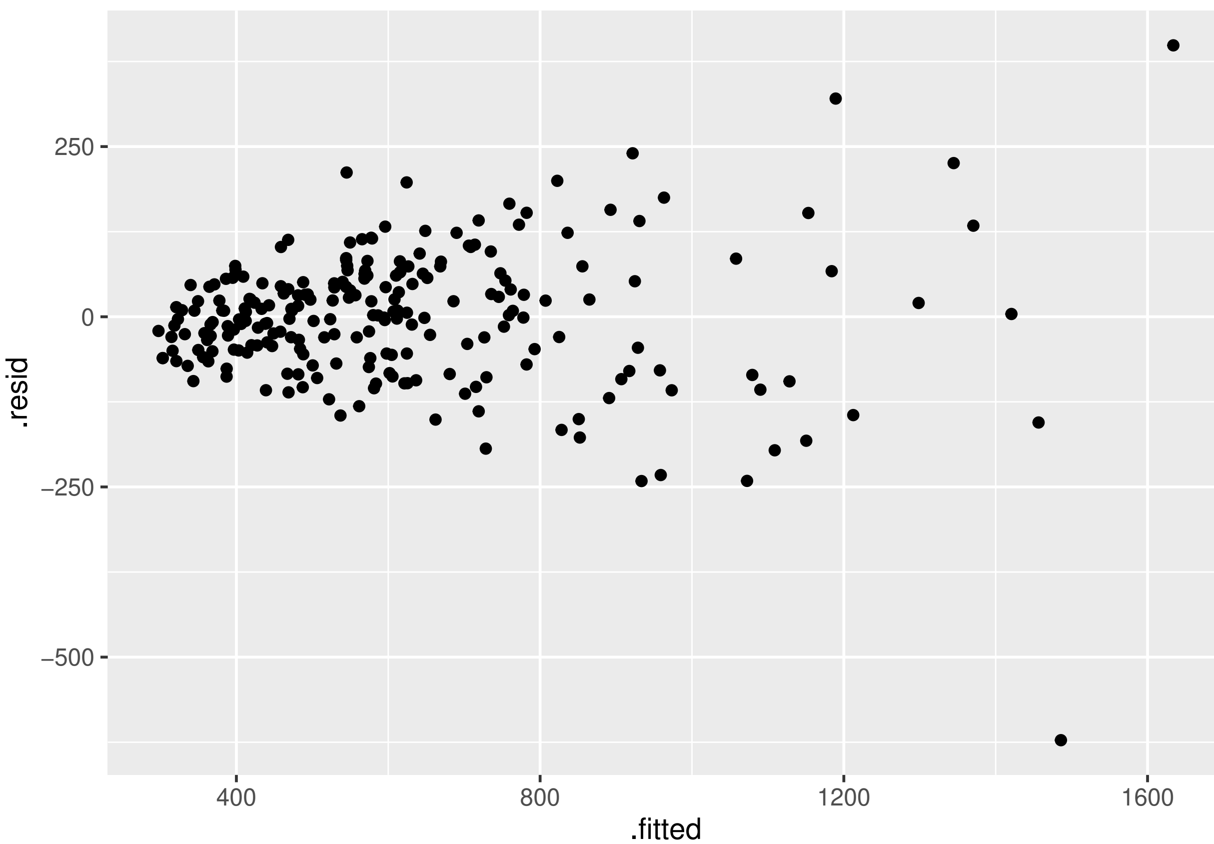

Figure 23: Food expenditure: residual plot 1

```
boxcox(foodexp ~ income, data = engel)
```

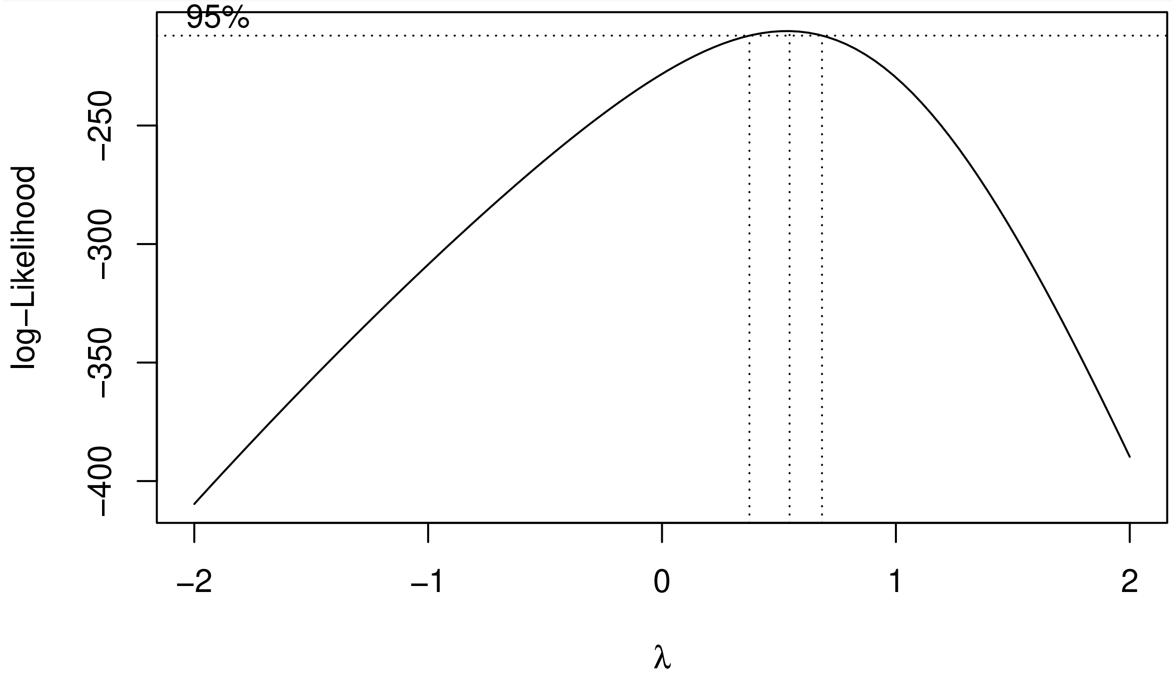

Figure 24: Food expenditure: Box-Cox

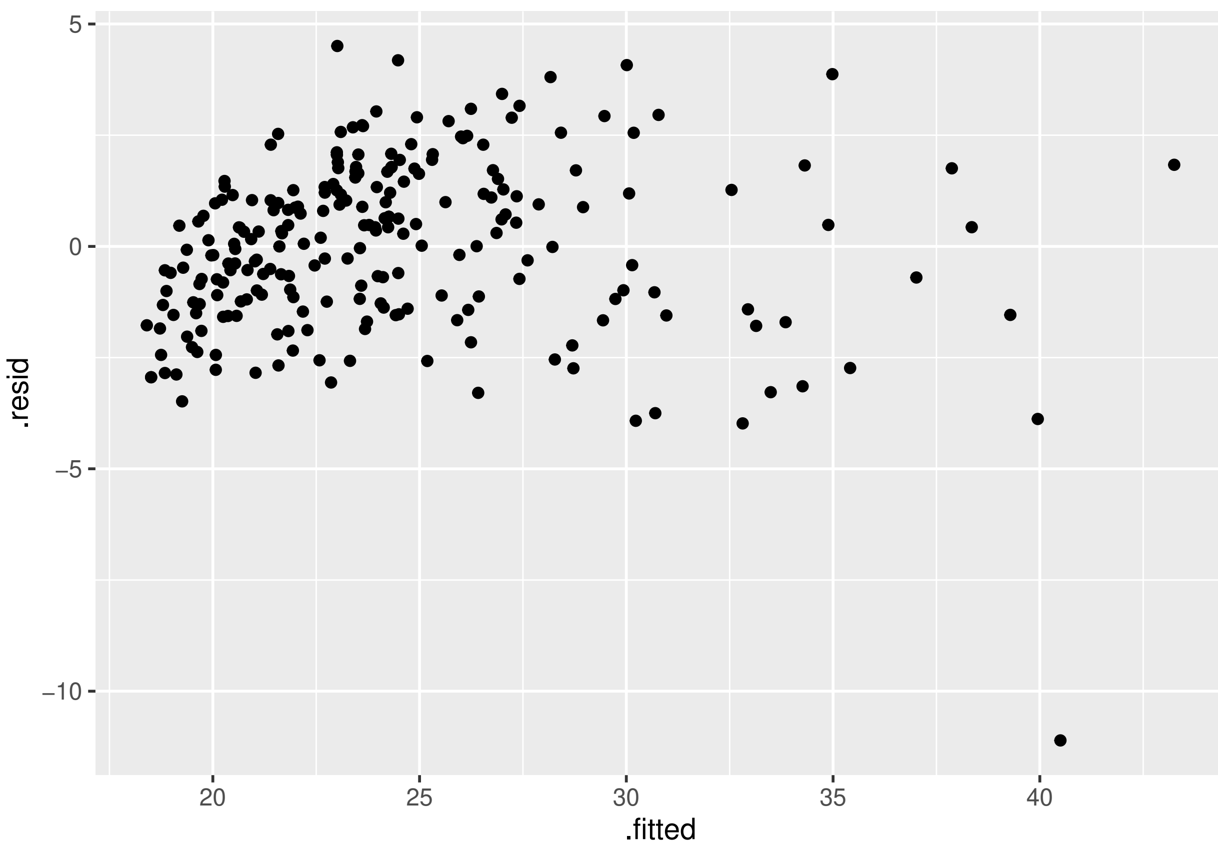

Figure 25: Food expenditure: residual plot 2

- `incomegp` income group (1=lowest, 5=highest)
- `house` security of housing tenure (1=rent, 2=mortgage, 3=owned outright)
- `children` number of children in household
- `singpar` is the respondent a single parent?
- `agegp` age group (1=youngest)
- `bankacc` does the respondent have a bank account?
- `bsocacc` does the respondent have a building society (credit union) account?
- `manage` self-rating of money management skill (high values=high skill)
- `ccarduse` how often did s/he use credit cards (1=never... 3=regularly)
- `cigbuy` does s/he buy cigarettes?
- `xmasbuy` does s/he buy Christmas presents for children?
- `locintrn` score on a locus of control scale (high values=internal)
- `prodebt` score on a scale of attitudes to debt (high values=favourable to debt (response variable)

Figure 26: Debt survey items

```
debt
```

```
## # A tibble: 304 x 13
##    incomegp house children singpar agegp bankacc bsocacc manage ccarduse cigbuy
##       <dbl> <dbl>    <dbl>   <dbl> <dbl>   <dbl>   <dbl>  <dbl>    <dbl>  <dbl>
##  1        3     3        0       0     4       1       0      5        2      0
##  2        5     2        2       0     2       1       0      5        3      0
##  3        3     3        0       0     4       1       0      4        2      0
##  4        4     2        0       0     2       1       0      5        2      0
##  5        4     2        0       0     2       1       0      4        2      0
##  6        2     1        1       0     4       1       0      4        1      0
##  7        2     3        0       0     4       1       0      5        1      0
##  8        2     3        0       0     4       1       0      5        1      0
##  9        2     3        2       0     4       0       1      4        2      0
## 10        2     2        2       1     3       1       0      4        1      1
## # ... with 294 more rows, and 3 more variables: xmasbuy <dbl>, locintrn <dbl>,
## #   prodebt <dbl>
```

Figure 27: Debt data (some)

```r
debt.1 <- lm(prodebt ~ ., data = debt)
summary(debt.1)
```

```
##
## Call:
## lm(formula = prodebt ~ ., data = debt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.95085 -0.46986 -0.01442  0.40263  1.87677
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.04642    0.31682  12.772  < 2e-16 ***
## incomegp     0.06463    0.03373   1.916 0.056336 .
## house       -0.05331    0.06751  -0.790 0.430378
## children     0.03813    0.03898   0.978 0.328749
## singpar      0.02054    0.17372   0.118 0.905984
## agegp       -0.10206    0.04761  -2.144 0.032899 *
## bankacc      0.06248    0.12123   0.515 0.606641
## bsocacc     -0.11198    0.08344  -1.342 0.180628
## manage      -0.12820    0.04556  -2.814 0.005231 **
## ccarduse     0.18779    0.05258   3.571 0.000415 ***
## cigbuy      -0.15448    0.08731  -1.769 0.077894 .
## xmasbuy      0.20147    0.11928   1.689 0.092298 .
## locintrn    -0.13942    0.04371  -3.190 0.001579 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6562 on 291 degrees of freedom
## Multiple R-squared:  0.2043, Adjusted R-squared:  0.1715
## F-statistic: 6.226 on 12 and 291 DF,  p-value: 8.916e-10
```

Using a dot on the right side of a model formula means “all the other variables”.

Figure 28: Debt data regression 1

```
debt.2 <- update(debt.1, .~. - singpar - bankacc - house - children - bsocacc)
summary(debt.2)
```

```
##
## Call:
## lm(formula = prodebt ~ incomegp + agegp + manage + ccarduse +
##     cigbuy + xmasbuy + locintrn, data = debt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.99736 -0.43552  0.00559  0.40031  1.81132
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.08091    0.29233  13.960  < 2e-16 ***
## incomegp     0.06025    0.03063   1.967 0.050125 .
## agegp       -0.13047    0.04143  -3.149 0.001805 **
## manage      -0.14141    0.04389  -3.222 0.001416 **
## ccarduse     0.18775    0.05149   3.647 0.000314 ***
## cigbuy      -0.13220    0.08560  -1.544 0.123579
## xmasbuy      0.22305    0.11479   1.943 0.052963 .
## locintrn    -0.14165    0.04330  -3.271 0.001198 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6554 on 296 degrees of freedom
## Multiple R-squared:  0.1926, Adjusted R-squared:  0.1735
## F-statistic: 10.09 on 7 and 296 DF,  p-value: 2.546e-11
```

`update` requires a model to update, and then how to update it. This one means "leave everything the same except take out the five explanatory variables listed."

Figure 29: Debt data regression 2

```
anova(debt.2, debt.1)
```

```
## Analysis of Variance Table
##
## Model 1: prodebt ~ incomegp + agegp + manage + ccarduse + cigbuy + xmasbuy +
##     locintrn
## Model 2: prodebt ~ incomegp + house + children + singpar + agegp + bankacc +
##     bsocacc + manage + ccarduse + cigbuy + xmasbuy + locintrn
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    296 127.14
## 2    291 125.31  5     1.836 0.8528 0.5134
```

Figure 30: Debt data: a test

```
ggplot(debt.2, aes(x = .fitted, y = .resid)) + geom_point()
```

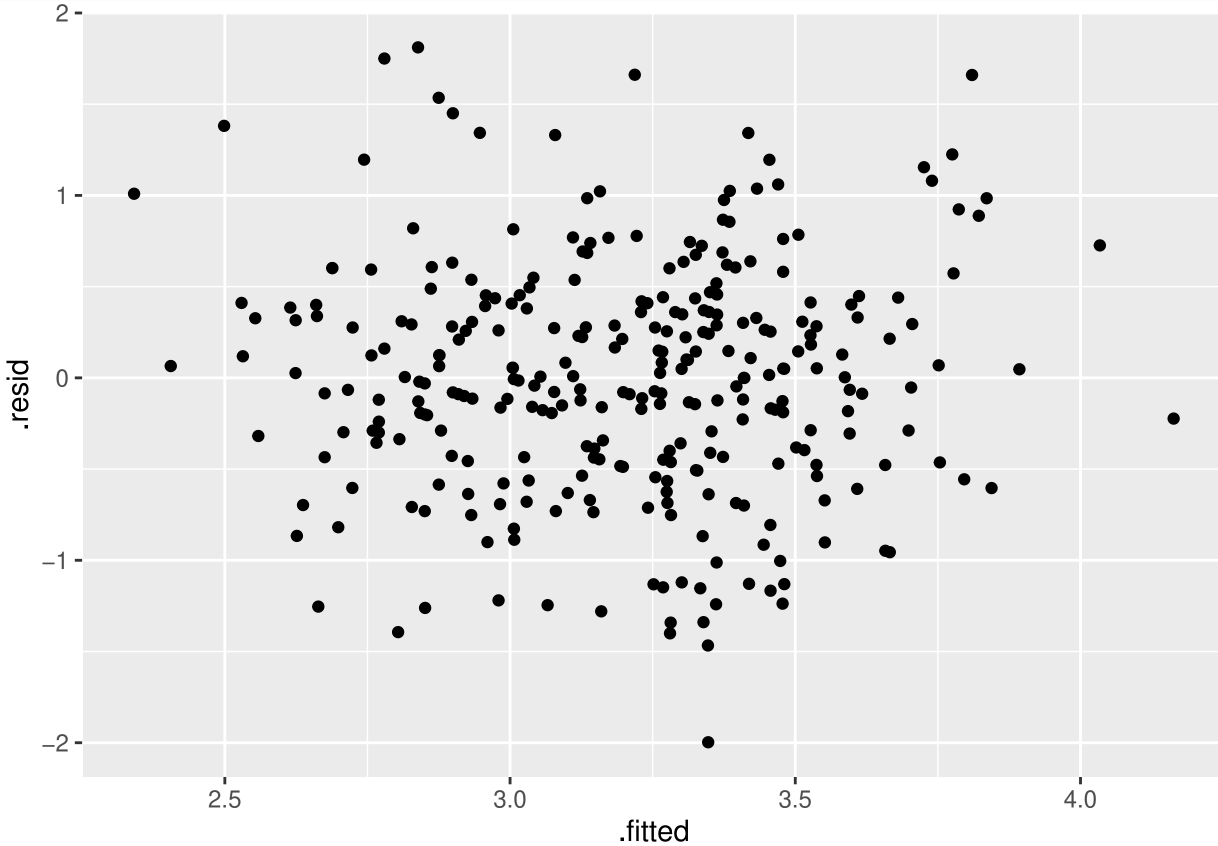

Figure 31: Debt data: residuals vs. fitted values from model `debt.2`

```
ggplot(debt.2, aes(sample = .resid)) +
  stat_qq() + stat_qq_line()
```

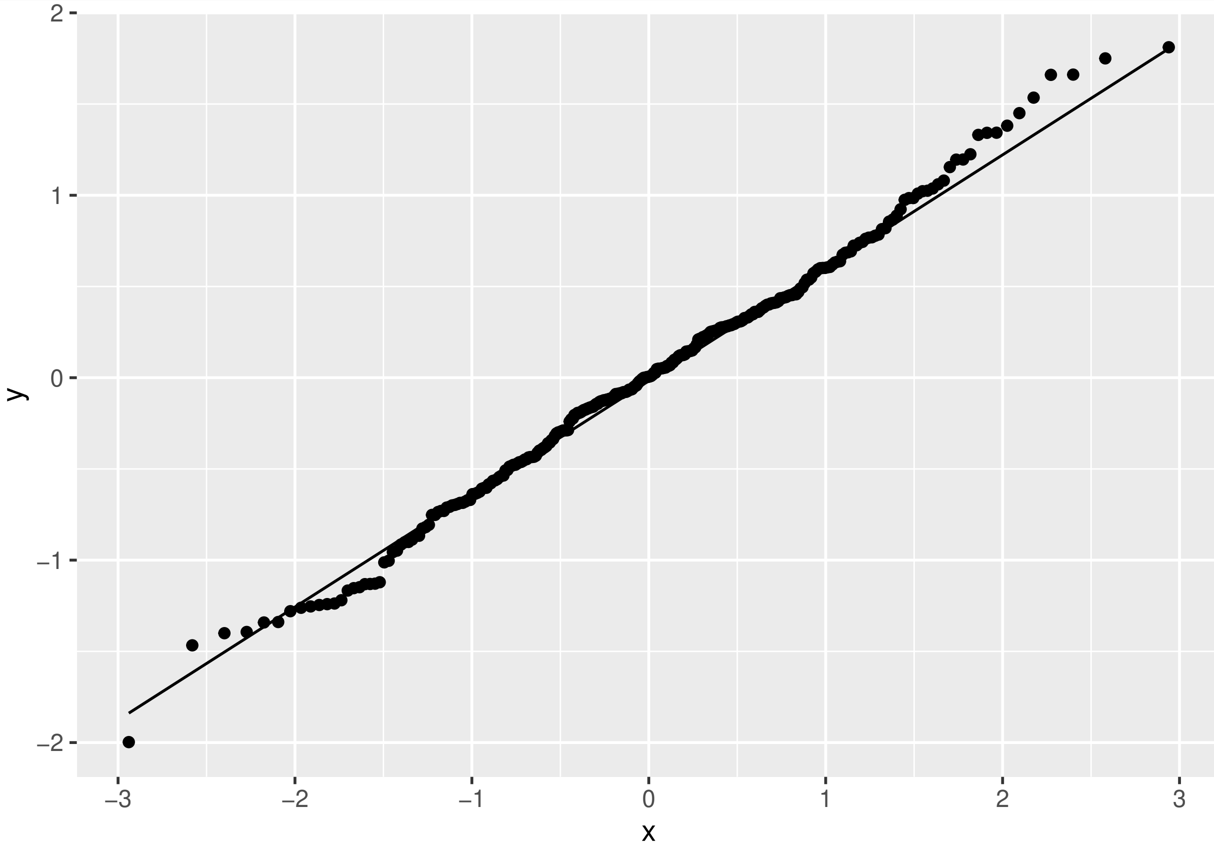

Figure 32: Debt data: normal quantile plot of residuals from model `debt.2`

```
w <- c(0.5, 5.4, 3.7, 13.8, 12.9, 4.0, 17.3, 6.6, 4.8, 2.5)
```

Figure 33: Observed data for estimating β by Bayesian methods

```
expo_fit
```

```
##  variable   mean median   sd  mad      q5    q95 rhat ess_bulk ess_tail
##      lp__ -31.94 -31.76 0.53 0.22 -32.80 -31.59 1.00     1379     1212
##      beta   0.16   0.16 0.04 0.04   0.11   0.24 1.00     1004     1154
```

Figure 34: Summary of posterior distribution of β