# University of Toronto Scarborough
# Department of Computer and Mathematical Sciences
# STAD29 (K. Butler), Final Exam
# April xxx, 2023

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 xxx 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.**

1. Two different treatment drugs are available for the treatment of cocaine addiction. These are called Desipramine and Lithium. In a study, 72 subjects (cocaine addicts seeking treatment) were randomly assigned to one of these treatments or to a placebo. The subjects were followed for a certain period of time, and a record was kept of whether they "relapsed", which means that they went back to using cocaine during that time period. This was recorded as `yes` or `no`. Some of the dataframe is shown in Figure 2. The aim of the study was to see whether there was a difference between the treatments, and if so, which one of the treatments made relapses less likely.

   (a) [3] What feature of the dataset makes logistic regression, as run with `glm`, a reasonable technique to use in studying these data?

   **My answer:**

   The outcome (response) variable `Relapse` is categorical, with two categories `yes` and `no`.

   Make sure you're clear about which variable is the response, that it is categorical, and what those categories are (or, at least, that there are only two of them).

   (b) [2] In Figure 3, a logistic regression model is fitted. How do you know that this model is predicting the probability that a subject *does* relapse, rather than the probability that the subject *does not* relapse? Explain briefly.

   **My answer:**

   This is an ordinary logistic regression with one observation per row, so the probability being modelled is that of the alphabetically second response category, `yes`: ie., that the subject *does* relapse.

   More precisely, the first category alphabetically is the baseline, and we predict the probability of the other one. But knowing it's the second category is enough here.

   (c) [2] In Figure 3, is there evidence of any difference among the treatments on the probability of relapse? In your answer, explain which part of the Figure you are basing your answer upon.

   **My answer:**

   The explanatory variable `Drug` is categorical with three categories, so we should use the `drop1` output at the bottom of the Figure. The null hypothesis is that the probability of relapse is the same for all three treatments, and with a P-value of 0.0054, this is clearly rejected. There is a difference in probability of relapse between the treatments.

   The two P-values in the `summary(cocaine.1)` output only compare the treatments shown with the baseline treatment `Desipramine`, which is therefore significantly different from both of `Lithium` and `Placebo`, but there is no indication of whether the two treatments shown are significantly different from each other (my guess is that they are not). One point only for approaching it this way.

   Extra: this is the same idea as for ANOVA; the equivalent to the $F$-test is significant, indicating

that there are some differences to find, but we do not have an equivalent to Tukey in this situation. The nearest we can get is "pairwise chi-squared tests", in the same spirit as the pairwise median tests as the followup to Mood's median test. See below for more discussion of this.

(d) [3] According to the evidence in Figure 3, which of the three treatments is the most effective? Explain briefly.

**My answer:**

The most effective treatment is the one that makes relapses *least* likely; that is, the one for which the (predicted) probability of `yes` is *smallest*.

In the `summary` output, the baseline `Drug` is Desipramine, which has an estimate of zero. This is much smaller than for either of the other two drugs, and so the probability of relapse is also much smaller for Desipramine than for the other two treatments.

It actually looks as if Lithium is not very much more effective than placebo.

Extra: Logistic regression really shines when the response variable is categorical but the explanatory variable is quantitative, like the rat poison example in lecture (where dose was quantitative). But it works equally well when the explanatory variable is categorical (or if you have a mixture of the two types). With categorical explanatory variables, you treat them the same way you would treat them in regular regression (that is to say, use `drop1` to test the categorical variable as a whole for significance, and remember that there is a baseline category that is not shown in the `summary`).

Another way of testing for an association between two categorical variables is a chi-squared test (as we saw in the preamble to Mood's median test in C32):

```
cocaine.2 <- with(cocaine, chisq.test(Drug, Relapse))
cocaine.2
```

```
##
##  Pearson's Chi-squared test
##
## data:  Drug and Relapse
## X-squared = 10.5, df = 2, p-value = 0.005248
```

The P-value from the chi-squared test is similar to, but slightly different from, the one we got in the logistic regression.

If you remember doing a chi-squared test by hand, you will (fondly?) remember working out the expected frequencies and doing a calculation to compare them with the observed ones. Here, these are:

```
cocaine.2$observed
```

```
##              Relapse
## Drug          no yes
```

```
##   Desipramine 14  10
##   Lithium      6  18
##   Placebo      4  20
```

```
cocaine.2$expected
```

```
##              Relapse
## Drug          no yes
##   Desipramine  8  16
##   Lithium      8  16
##   Placebo      8  16
```

This shows that out of the 24 subjects in each treatment group, 16 were expected to relapse (under the null hypothesis), but only 10 of the Desipramine subjects actually did, meaning that Desipramine was more effective. Out of the Lithium patients, 18 relapsed, which was not much better than for the control patients (20 relapsed). These conclusions are very much the same as the ones we drew from the logistic regression, as we would expect.

In the same spirit as Mood's median test, we could follow this up with "pairwise chi-squared tests" to compare each pair of treatments. In the same way as that, we have to recognize that we are doing three tests at once, so we multiply each P-value by three (Bonferroni correction) before drawing a conclusion. Here's the first one:

```
cocaine %>%
  filter(Drug != "Desipramine") %>%
  with(., chisq.test(Drug, Relapse))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  Drug and Relapse
## X-squared = 0.12632, df = 1, p-value = 0.7223
```

This one is a comparison between `Lithium` and `Placebo`. It was easier to say which treatment I wanted to *exclude*. The other code thing to talk about is in the `with` line, where the "dot" means "the dataframe that came out of the previous step", that is, the one without the `Desipramine` patients in it.

`Lithium` is (as we guessed) no better than `Placebo`.

The other two are a case of copying and pasting:

```
cocaine %>%
  filter(Drug != "Lithium") %>%
  with(., chisq.test(Drug, Relapse))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  Drug and Relapse
```

```
## X-squared = 7.2, df = 1, p-value = 0.00729
```

This compares `Desipramine` and `Placebo`. These are significantly different, even after multiplying the P-value by 3; Desipramine is actually significantly *better* at preventing relapses.

Finally:

```
cocaine %>%
  filter(Drug != "Placebo") %>%
  with(., chisq.test(Drug, Relapse))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  Drug and Relapse
## X-squared = 4.2, df = 1, p-value = 0.04042
```

The two real treatments against each other. Once you adjust the P-value for doing three tests at once, it is no longer significant ($0.04 \times 3 = 0.12$). Thus the only significant difference is between Desipramine and Placebo, with Lithium occupying an indefinite middle ground.

It might have made more sense for you to see the chi-squared output for these data, but I wanted a question that would test your understanding of logistic regression, so that's what you got.

2. Samples of worsted yarn (wool) were stretched until they broke, and the time until they broke was recorded. Specifically, there were two factors that might affect how long it took for a sample of worsted to break: the length of the piece of worsted (250, 300, or 350 mm), and the "amplitude of the loading cycle". The worsted is stretched for a certain period of time before being released and then stretched again. Each sample was repeatedly stretched for one of three periods of time (8, 9, and 10 minutes), and the response variable `cycles` is the number of stretch-release cycles that the sample of worsted underwent before it broke. A higher value of `cycles` thus indicates that the yarn is stronger.

   The dataframe is shown in Figure 4. The explanatory variables mentioned above are in columns named `len` and `amp` (respectively); the response variable is called `cycles`. There is an additional column called `load` in the Figure that we ignore in this question.

   This was a designed experiment, with the values of `len` and `amp` chosen by the experimenter. We therefore want to treat `len` and `amp` as categorical (even though their values are actually numbers); we don't want to assume a linear relationship with their numerical values.

   (a) [2] A grouped boxplot is shown in Figure 5. From this plot, why would you expect to see a significant interaction between length and amplitude?

   **My answer:**

   The number of cycles is always highest for amplitude 8, but it is *much* higher for amplitude 8 when length is 350, compared to the other lengths: that is to say, the *amount* by which the number of cycles is higher for amplitude 8 is itself higher for length 350 than for the other

lengths.

Say something that gets at the idea of the effect of amplitude on the number of cycles depending on the length: "the effect of one explanatory variable depends on the level of the other", as it applies here. Be as specific as you can.

This is not an interaction plot, so you don't immediately have the idea of "the lines are not parallel" to take advantage of. If you want to try, you need to imagine lines going through the medians of the boxes, joining the three boxes of the same colour (which you will have to explain). The line joining the three red boxes goes up faster than the other two lines joining the boxes of the other colours.

(b) [2] An analysis of variance is shown in Figure 6. What do you conclude from this Figure?

**My answer:**

Simply, there is a significant interaction between length and amplitude, and *stop there.*

This is exactly what the boxplots said.

(c) [4] Figure 7 shows two parts of a simple-effects analysis. What specifically do you conclude from this Figure? (There are two parts in the Figure, labelled SE1 and SE2. You need a conclusion from each.)

**My answer:**

SE1 says that when length is 350, there is a significant effect of amplitude, and the Tukey below it says that `cycles` is significantly different (larger) for an amplitude of 8 than for the other two amplitudes (which are not significantly different from each other).

SE2 says that when length is 250, there is *not* a significant effect of amplitude at all (P-value 0.0538), and therefore we should *not* look at the Tukey below it.

(d) [2] Why was it not surprising, given what we have seen so far, that your two conclusions in the previous part were different from each other? Explain briefly.

**My answer:**

The two conclusions in the previous part said that whether or not there is an effect of amplitude depends on the value of length. Before looking at the simple effects, we found a significant interaction between amplitude and length, which means that the effect of amplitude depends on length. The simple effects showed that this is indeed true, and more specifically *how* it is true.

(e) [2] Why does the boxplot in Figure 5 cause you to doubt the analysis we have just done, and why in particular does it suggest that a transformation of `cycles` like square root or log might be useful?

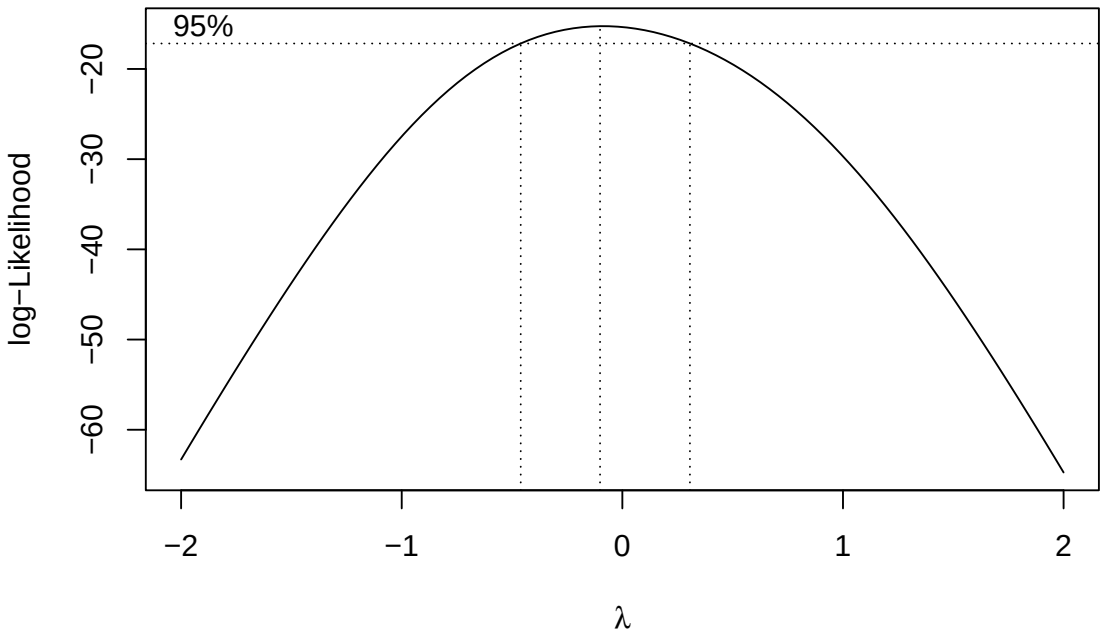
**My answer:**

Each boxplot is based on only three observations ($3 \times 3 = 9$ treatment combinations, and 27 observations altogether) so we don't learn much about normality, but there seems to be a consistent pattern of unequal spread, which should make us doubt the analysis we just did. (We are in standard ANOVA territory here, with our assumptions being normality and equal spreads.)

The reason that such a transformation might help is that the boxes with bigger spreads also have bigger medians (that is to say, the larger values are more spread out). Doing a transformation like log or square root will bring the bigger values down further, which means it will also make them less spread out.

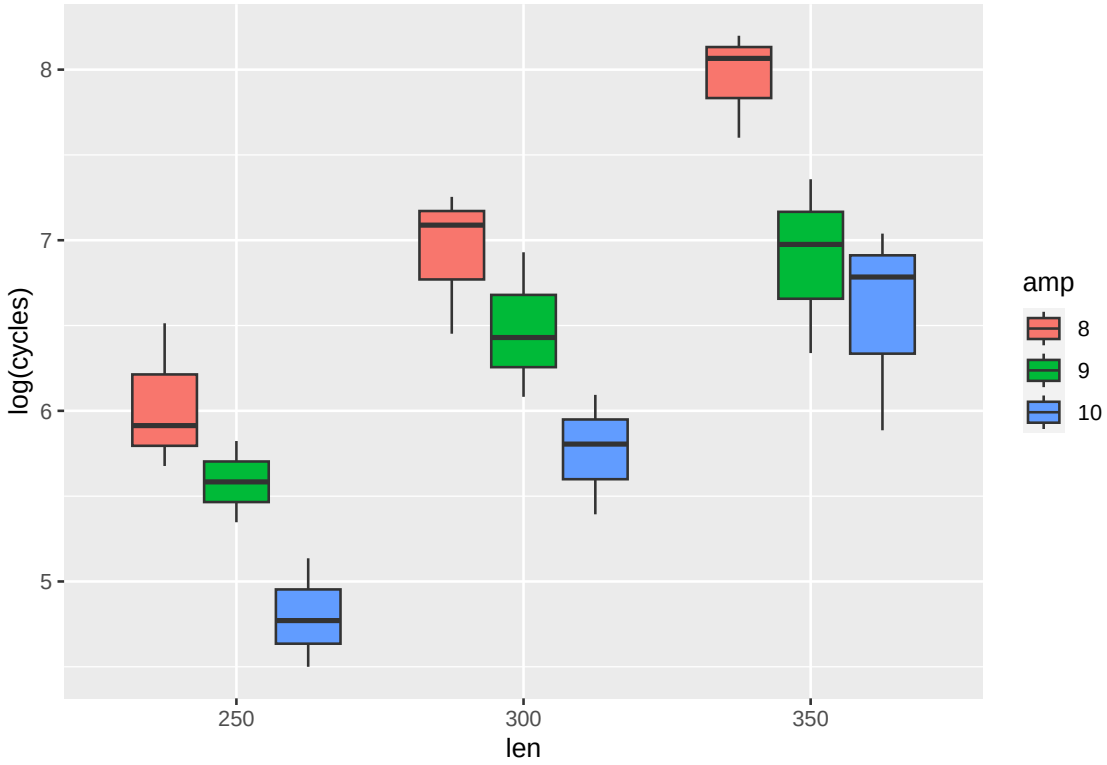
Extra: this dataset came from Box and Cox's original paper on their transformation, which dates from 1964. They actually did a three-way analysis including `load`, which we will not worry about. A Box-Cox analysis for these data according to our analysis is shown here:

```
boxcox(cycles ~ len * amp, data = Wool)
```




As you see, the log transformation is very much suggested here (the confidence interval for $\lambda$ doesn't stretch up as far as 0.5, square root, or down as far as $-0.5$, one over square root). Does taking logs improve the boxplot?

```
ggplot(Wool, aes(x = len, y = log(cycles), fill = amp)) + geom_boxplot()
```




Very much so. Given that each boxplot is based on only three observations, we seem to have done an excellent job of getting the spreads to be about the same.

Box and Cox also make a point in their paper that with a good transformation often comes simpler structure:

```
wool.2 <- aov(log(cycles) ~ len * amp, data = Wool)
summary(wool.2)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## len          2 12.516   6.258  36.125 4.99e-07 ***
## amp          2  7.167   3.584  20.687 2.16e-05 ***
## len:amp      4  0.401   0.100   0.579    0.682
## Residuals   18  3.118   0.173
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The interaction is no longer anywhere near significant, so it can come out:

```
wool.3 <- aov(log(cycles) ~ len + amp, data = Wool)
summary(wool.3)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## len          2 12.516   6.258   39.12 5.69e-08 ***
## amp          2  7.167   3.584   22.40 4.94e-06 ***
## Residuals   22  3.519   0.160
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(wool.3)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = log(cycles) ~ len + amp, data = Wool)
##
## $len
##               diff      lwr      upr     p adj
## 300-250 0.9183326 0.444699 1.391966 0.0002053
## 350-250 1.6647683 1.191135 2.138402 0.0000000
## 350-300 0.7464357 0.272802 1.220069 0.0018529
##
## $amp
##             diff       lwr        upr     p adj
## 9-8  -0.6552099 -1.128844 -0.1815762 0.0058466
## 10-8 -1.2617320 -1.735366 -0.7880984 0.0000029
## 10-9 -0.6065222 -1.080156 -0.1328885 0.0106437
```

Now we have a consistent effect of amplitude (8 is higher than 9 is higher than 10) that applies over all lengths, and a consistent effect of length (the longer the better) that applies over all amplitudes. No need to look at simple effects any more! This is what Box and Cox meant by "simpler structure": if you can find a good scale for the response, which might not be the scale you originally measured it on, you may be able to make things easier to understand, as well as better matching the assumptions of your analysis.

3. Salmon are born in fresh water (rivers) before travelling to salt water (the sea) before returning to fresh water to have their young. Some of the 100 salmon in our dataset were born in Alaskan rivers, and some in Canadian rivers; some were male, and some were female. Do these salmon grow differently? One way to measure this is to look at the diameter of growth rings for the first year of life in the river, and also for the first year of life in the sea. Some of the data are shown in Figure 8. The columns are respectively the `Gender` of each salmon (1 is female, 2 is male), the growth ring diameter in the river (`Freshwater`), the growth ring diameter in the sea (`Marine`), where the salmon was born (`Origin`), and the combination of origin and gender, for use later.

   The growth ring diameters are measured in hundredths of an inch.

(a) [2] Why would MANOVA be suitable for analyzing these data?

**My answer:**

There are two (quantitative) response variables, `Freshwater` and `Marine`, rather than only one.

Extra: this presentation (it's a link) talks about how salmon scales have rings like the rings of a tree; the diameter of these rings is what we are measuring here. Experts can use these rings to determine a salmon's age, which type of salmon it is, and other things.

(b) [2] What code would create a suitable response variable for a MANOVA?

**My answer:**

The base R way:

```
response <- with(salmon, cbind(Freshwater, Marine))
head(response)
```

```
##      Freshwater Marine
## [1,]        108    368
## [2,]        131    355
## [3,]        105    469
## [4,]         86    506
## [5,]         99    402
## [6,]         87    423
```

or the Tidyverse way:

```
salmon %>% select(Freshwater, Marine) %>%
  as.matrix() -> response
head(response)
```

```
##      Freshwater Marine
## [1,]        108    368
## [2,]        131    355
## [3,]        105    469
## [4,]         86    506
## [5,]         99    402
## [6,]         87    423
```

Either is good. Call the response matrix whatever you like. There is no need to display it. I just wanted to demonstrate for you that my code worked.

In total, `response` has 100 rows, so in principle, you should display only some of it (`response` is a `matrix` rather than a dataframe, so displaying it will display *all* of it unless you do something to stop that happening.)

(c) [4] A MANOVA analysis is shown in Figure 9. There are three parts to the analysis. Explain briefly

why I did the analysis the way I did, and say what your final conclusion from the Figure is in the context of the data. (Note that `response` in the Figure is the response variable you created earlier.)

**My answer:**

First, I see how our combined response depends on the gender and origin of the fish and their interaction. Then, the interaction is not significant so I remove it. There is then a significant main effect of origin but not gender, so I remove gender from the model as well. Two pretty straightforward points for getting this far.

This means that the freshwater growth ring diameter or the marine growth ring diameter, or some combination of them, depends on where the salmon was born. Say this much, with suitable vagueness. The other two points.

We cannot say any more about the dependence yet, but this much we know.

(d) [2] Why is a discriminant analysis worth considering at this point?

**My answer:**

We want to know *how* the two growth ring diameter values depend on origin (and maybe gender, but the analysis makes it seem that only origin will be important). The MANOVA says that they do depend on origin, but not how they do.

(e) [2] A discriminant analysis is shown in Figure 10. Why are there *two* linear discriminants, and why is only one of them apparently important in distinguishing the origin-gender combinations?

**My answer:**

There are two quantitative response variables and four origin-gender combinations, and the smaller of 2 and $4 - 1$ is 2.

Only one of them seems to be important in distinguishing the groups because the proportion of trace for LD1 is large and for LD2 is very small. This indicates that LD2 probably does almost nothing to distinguish the groups.

One point each.

(f) [2] What values of the original variables would make LD1 *large*? (You may assume that neither coefficient is close to zero.)

**My answer:**

The coefficient of `Freshwater` is positive, and the coefficient of `Marine` is negative, so LD1 will be large if `Freshwater` is large and `Marine` is *small*.

Extra: you can guess from the group means that large LD1 scores will tend to go with Canadian

salmon, since they tend to have a larger value of `Freshwater` and a smaller value of `Marine`.

(g) [4] Some further discriminant analysis is shown in Figure 11, and a plot of the scores of each salmon on the first two linear discriminants is shown in Figure 12. From the plot, would you say that either the origins or the genders of the salmon are distinguishable? Explain briefly.

Note: there are four colours on the plot: black, orange, blue, and green. If you cannot tell all four colours apart, ask an invigilator to help you identify some points. You might like to think first about which points would be helpful to identify. I tried to find a colour-blind-friendly collection of colours (see Note above the Figure).

**My answer:**

The Alaskan salmon are the black and orange points, mainly on the left of the graph (small LD1 score). The Canadian salmon are the blue and green points, mainly on the right (large LD1 score). So the salmon are distinguishable by origin.

In terms of gender, though, the males and females are all mixed up, both for the Alaskan and Canadian salmon. So the genders are not distinguishable at all.

(h) [2] Figure 13 shows a table of actual and predicted genders and origins for the salmon. How does this table support your answer to the previous part? Explain briefly.

**My answer:**

Origin: very few Alaskan salmon are classified as Canadian (or vice versa).

Gender: a large number of 1 (female) are classified as 2 (male) and vice versa.

That is to say, here too it is easy to distinguish salmon by origin, but it is hard to tell males and females apart, at least from the measurements we have.

(i) [3] Taking the MANOVA and the discriminant analysis together, what do you conclude overall about how `Freshwater` and `Marine` differ according to the origin and gender of the salmon? (You may wish to consider what you would write in the Conclusion of a report about these data.)

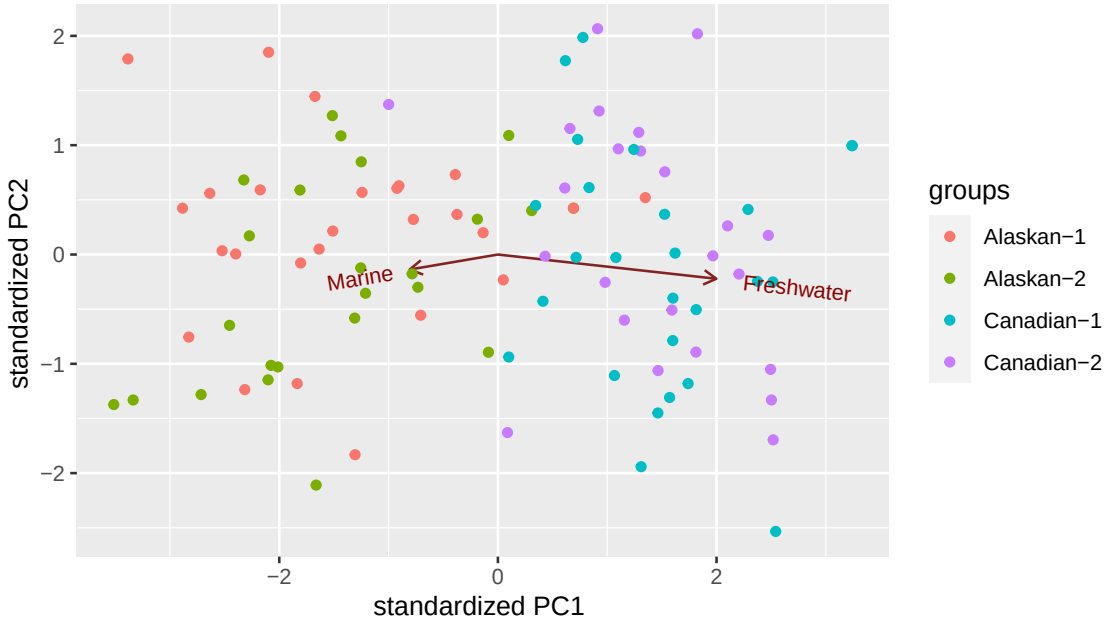
**My answer:**

- The MANOVA says that `Freshwater` and `Marine` differ only by the origin of the salmon, and not by their gender.
- This is supported by the discriminant analysis, which says that the gender of the salmon is difficult to distinguish, but that the origin of the salmon says a lot about their growth ring diameter values.
- Specifically, the Canadian salmon have a large (positive) LD1 score, which means that they have a larger value of `Freshwater` and a smaller value of `Marine`, compared to the Alaskan salmon.

The three points are roughly for those three bullet points, either stated or implied by other things you write. I expect that I will need to be flexible grading this, but I would like to see a conclusion from the MANOVA, and something fairly specific about how the discriminant analysis adds to what the MANOVA tells you.

Extra: when I was planning this question, I also drew a biplot, but I didn't give you that to look at, because it made the rest of it too easy:

```
ggbiplot(salmon.4, groups = salmon$combo)
```




from which you see easily that the Canadian salmon tend to have larger values of `Freshwater` and smaller values of `Marine`, and that the distinction happens left and right along LD1, with LD2 not distinguishing any groups.

4. Thirty-four individuals took part in a study of self-esteem. Each subject was randomly assigned to one of three treatments: diet, diet plus exercise, and a control group with no special instructions for diet or exercise. Each subject had their self-esteem assessed 1, 2, and 3 months after the start of their treatment. The data are shown in Figure 14. The columns `se1` through `se3` are the measurements of self-esteem at the three times.

   (a) [2] One of the study authors tells you that this is a two-way ANOVA with two factors treatment and time. Why do you think this is *not* an appropriate way to analyze the data?

   Note: I am not asking about the arrangement of the data; your answer should explain why the data, *even if suitably rearranged*, should not be analyzed as described by the study author.

**My answer:**

The proposed analysis assumes that each observation is independent of the others, which is not the case because the observations made on the same subject are likely to be correlated. Or, say that the data consist of three repeated measurements for each subject, so that a repeated measures analysis is the appropriate one.

(b) [2] An interaction plot is drawn, using the code shown in Figures 15 and 16. Why is the code in Figure 15 necessary?

**My answer:**

The dataframe in Figure 14 is "wide", with all the measurements for each subject on one row. To make a plot, we need all the self-esteem values in *one* column, with a second column saying what subject and time each one is for. That is to say, the plot needs "long" data, and the `pivot_longer` makes that.

(c) [2] Does the plot in Figure 16 suggest that there will be an interaction between treatment and time? Explain briefly why or why not.

**My answer:**

The three lines do not seem to be parallel: for the two real treatments (green and blue), self-esteem on average goes down and then up again (finishing up higher than it started), but for the control treatment, self-esteem is more or less constant over time. You could also say that the green and blue lines are not parallel either, since the blue line goes down and up further than the green one, but whether that difference is significant is not so clear.

Because the lines are not parallel, we would expect to see an interaction between treatment and time.

(d) [4] A suitable analysis is shown in Figure 17. Using the output shown, what do you conclude from this Figure? Be specific about where you are drawing your conclusions from. (There are usually four parts to this output, with the Multivariate Analysis shown first. Draw an appropriate conclusion based on the results shown here, which are displayed in the same order as on the complete output.)

**My answer:**

I actually didn't show you the Multivariate Tests that would normally appear on the (long) output that was too long to fit on one page. (I found out how to show you only parts of it.)

So, test for sphericity and do the appropriate analysis based on what you find.

The first thing is thus Mauchly's test in the middle of the output, labelled in a comment. This is significant (P-value 0.014), so we reject sphericity, and so we ignore the Univariate Tests at the top. In the bottom table, the second column of numbers is the P-values for

the Greenhouse-Geisser adjustment, and the fourth column of numbers is the same for the Huynh-Feldt adjustment. Use either.

Thus, what you do is to look at one of the adjusted P-values for the treatment-by-time interaction: 0.011 for Greenhouse-Geisser, 0.0099 for Huynh-Feldt. Quote the P-value you are looking at, but either way there is a significant interaction between treatment and time, and the pattern you saw on the interaction plot is real, not just chance. (I have no preference for which of the two adjustments you use: you can see that the P-values are very close, but you need to use one of them because sphericity fails.)

Having found a significant interaction, *stop there*: the main effects have nothing to tell you when there is a significant interaction. For a repeated measures analysis, if the interaction is *not* significant, then you *can* go ahead and look at the main effects, because there is no way to "take out" time, but that is not the situation here.

Extra: If you look back at the univariate tests (which I said to ignore), you see that these slightly overstate the significance of the interaction; the corrected P-value is not quite as small as the 0.006 we see there. But sphericity, though rejected, is not resoundingly rejected (P-value of 0.014 rather than smaller), so the P-values in the univariate tests are not far different from the adjusted ones.

(e) [2] How might you gain some insight about whether the diet plus exercise group consistently ends up with higher self-esteem than the diet-only group?

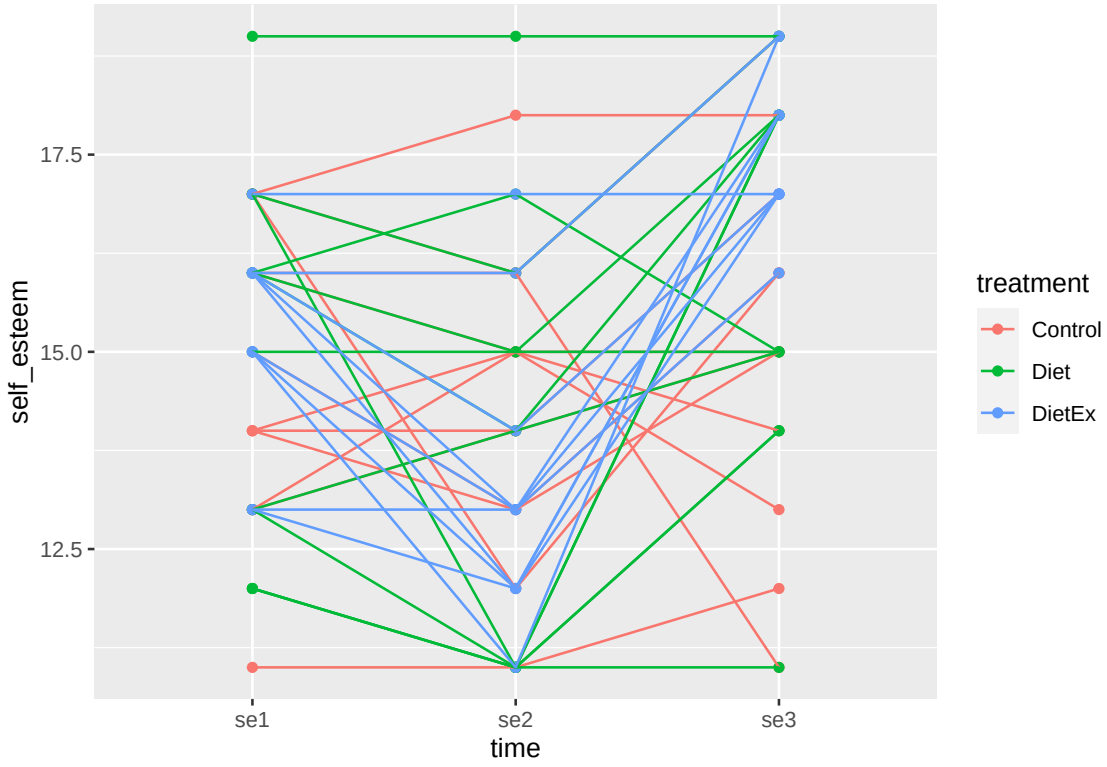
**My answer:**

The word "consistently" is meant to make you think about variability, which the interaction plot does not show, but the spaghetti plot does. So the answer I am thinking of is to suggest drawing a spaghetti plot, and then saying to look at it and see how consistently the subjects in the diet plus exercise group ended up highest after 3 months.

I am prepared to entertain other ways of getting at this. For example, you might say to calculate means and standard deviations of self-esteem scores over time (by treatment), and then check to see whether the standard deviations are "small". (You won't have any numbers, so this is about as far as your description can go.) Or you can suggest drawing grouped boxplots (with time and treatment as the categorical variables) and look at how tall the boxes are, which gets to the same thing. Say something sensible.

Extra: I have the data, so I can try some of these things out. I will be using the dataframe I called `xx` in Figure 15, which is the long-format version of our data. First the spaghetti plot:

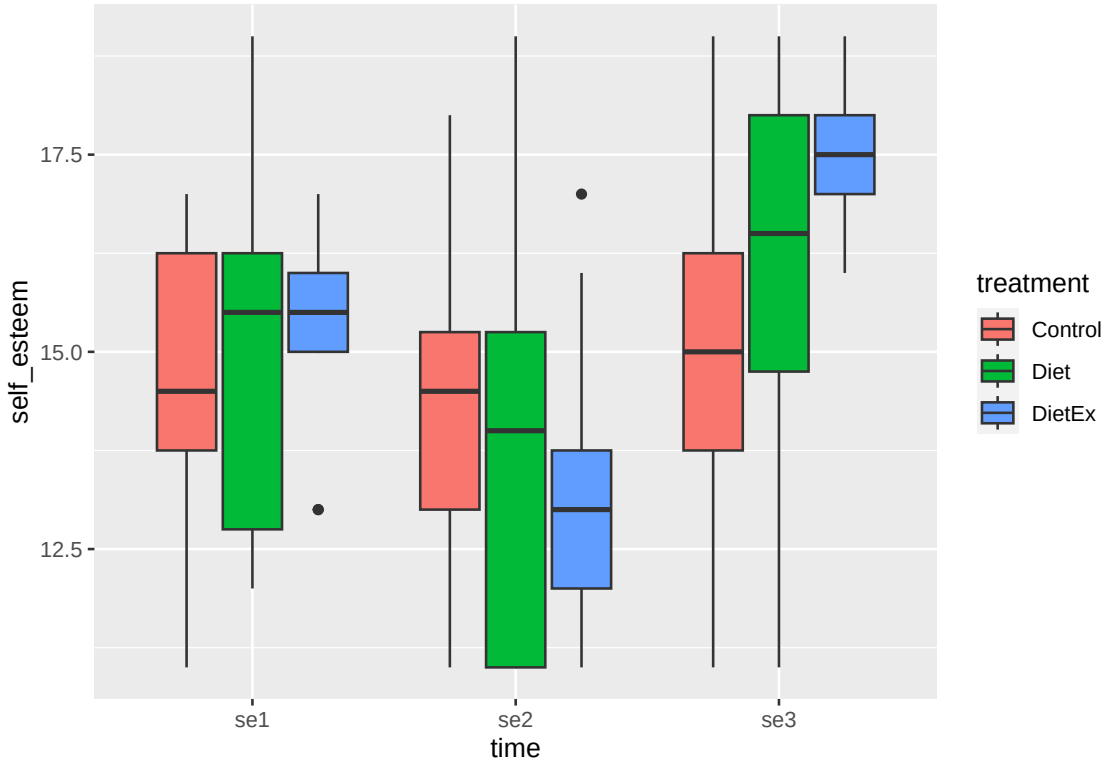
```
ggplot(xx, aes(x = time, y = self_esteem, colour = treatment, group = subject)) +
  geom_point() + geom_line()
```




It seems to me that there is a fair bit of variability in the down and up patterns of the subjects on the two real treatments, and it is not clear that the diet-plus-exercise people ended up highest consistently. Most of the blue traces *do* end up at the top, but there are some green ones mixed up in there as well. So my guess is that there is not evidence here that diet plus exercise is better than diet alone. Having said that, you might see that most of the blue traces ended up higher than they started, which is less true of the green ones. I didn't put this in a question for you, because the kind of answers I got would be very diverse and would take me too long to grade!

The other plot I suggested is a grouped boxplot, with time and treatment as groups. This is strictly not the right thing, because it loses the connection between the measurements that belong to the same subject (which is the same objection as you raised in part (a)), but there is some insight to be had here anyway. I like time on the $x$-axis for these:

```
ggplot(xx, aes(x = time, y = self_esteem, fill = treatment)) + geom_boxplot()
```




Once again, there seems to be too much variability to say confidently that the blues end up higher than the greens. The diet group seem to have a lot of variability in their results all the way along. You might say that the diet treatment sometimes works well (in terms of self-esteem), but the diet plus exercise treatment more *consistently* works well. There are actually two sources of variability here: one is the usual random variation, but one reason the green boxes are so tall is that there happened to be a lot of variability among the subjects in the `Diet` group. This is what I meant about losing the connection between the measurements; what really matters is whether the subjects in one group ended up consistently higher at the end *than they were at the beginning*, that is, comparing the measurements on the same subject. This is why a spaghetti plot will (potentially) offer more insight.

For the first point in this question, I'm looking for a sensible suggestion of something to do (a graph or some numerical summaries), and for the second point, I'm looking for some discussion of what to do with it: how you would know from your graph/summary whether the diet plus exercise group consistently ended up highest.

5. In 1969, cars were much less reliable than they are now. A car magazine kept track of the service records of 33 car models that were popular in 1969, and rated each model on the 13 items shown in Figure 18. The data are shown in Figure 19. A `+` in the dataframe means that this model of car needed repair for the item shown more often than average, and a `-` means that repair was needed less often than average. Our aim is to cluster the car models into groups of similar ones.

Some of the car names end in a number (which is often the number of cylinders in the engine), and some of them end in Full, which is short for "Full Size". These last were big cars that consumed a lot of gas (which in those days was not considered to matter very much).

(a) [2] A dissimilarity measure that is often used for data like this is the "simple matching coefficient". This is defined as the fraction of items on which two car models disagree, out of the 13 items that they might disagree on. The simple matching coefficient for the AMC Ambassador 8 and the Buick Special 6 is 0.385, to three decimals. Verify that this value is correct, showing your process.

**My answer:**

The two cars you are comparing are the top two lines of the dataframe in Figure 19, to make the verification easier for you.

These two car models disagree on five items (brakes, rattles & squeaks, rear axle, rust, other) out of 13, so their simple matching coefficient is

```
5/13
```

```
## [1] 0.3846154
```

Use your calculator for this, or if you don't have a calculator, get as far as 5 out of 13 and convince yourself that this must be about 0.385. I will accept an answer of "5 divided by 13" even if you don't do the calculation.

(b) [1] I computed the simple matching coefficient for each pair of car models (not shown), and ran a hierarchical cluster analysis using Ward's method, shown in Figure 20. What does the code `rect.hclust` shown in the Figure do?

**My answer:**

It draws the red rectangles around the number of clusters shown. That's it.

(c) [2] Why do you think I decided to use two clusters? Explain briefly.

**My answer:**

There are two clusters for a large part of the vertical distance down the dendrogram. After two clusters splits into more, it continues to split frequently, so if two is not a good number of clusters, it is not clear what *is* a good number.

Say some appreciable fraction of that.

(d) [3] I counted up, for each car model, the number of the 13 items that are above average. This calculation is shown in Figure 21, based on the original dataframe in Figure 19, with the column `n` in the dataframe `problems` indicating the number of items for each model that were `+` rather than `-`.

Some additional computation, and a boxplot, is shown in Figures 22 and 23. I have shown you the dataframe `clusters`. Describe in words what the three lines of code in Figure 23 are doing, in such

a way that somebody who knows about cluster analysis and some basic statistics but who does *not* know R can understand your description. The line beginning `##` above the plot is output, not code.

You need to be familiar with all the code in Figure 23; you are *not allowed* to ask an invigilator about it.

**My answer:**

- Line 1 makes a dataframe containing the car models with both the cluster each one belongs to and the number of "problem" items (that were denoted with a `+`) for that model. (You should avoid using the word "join" here, because your reader may not know about that; you can use the words "look up" if you want, but you should convey some understanding of what the dataframe created in this line actually contains.)
- Line 2 turns the (numeric) clusters into categories (a categorical variable).
- Line 3 makes a boxplot of the values of the number of "problem" items for the cars in each cluster. (This is a giveaway!)

Extra: you can easily hand-confirm that some of the values in `n` are what you think they are, for example that there are exactly two `+` values in the entire first row of Figure 19, the AMC Ambassador.

(e) [2] In a few words, explain what you conclude from the graph at the bottom of Figure 23, in the context of the data.

**My answer:**

The cars in cluster 1 have a better repair record than the cars in cluster 2.

That is the best summary I can think of. You can be a bit more precise about "repair record", something like "the cars in cluster 1 have a below-average number of repairs on more items than the cars in cluster 2". I wanted you to be able to read through my code and understand what I was doing; as I said in the question, the code itself should be familiar, and your task is to show enough understanding of how the code goes together to be able to describe my process. (In the workplace, you can expect to inherit code from other people, and you'll need to be able to see quickly what it is doing.)

Extra: most of the cars in this dataset are American, and in fact most of them were made by the "big three" American automakers (Ford, General Motors (AMC, Chevrolet, Buick), Chrysler). There are a few cars made elsewhere, eg. Mercedes, Porsche, Peugeot, Renault; these are all in the "better repair record" cluster 1. Make of that what you will.

A more nuanced analysis might reveal exactly how the better repair record shows up (eg, do the cars in cluster 2 tend to need a lot of *certain kinds* of repairs?). In one sense, looking at the total number of plusses is an oversimplification, but possibly a useful one.

Historically, 1969 was just before the time when Japanese cars started to make their mark in North America. There was an oil crisis in the mid-1970s when fuel became much more expensive, and suddenly there was a big demand for small, reliable, fuel-efficient cars such as the ones being made in Japan.

6. The city of Boston, Massachusetts, had 506 distinct census tracts in 1970. For each one, 14 variables were measured, as shown in Figure 24. Some of the dataframe is shown in Figure 25. We will be doing a principal components analysis.

   (a) [3] Figure 26 shows a scree plot. What do you conclude from this plot? Explain briefly.

   **My answer:**

   Look for a convincing elbow on the scree plot that is reasonably far down the mountain. I see elbows at 2, 4, and 9. I think the one at 2 is a bit far up the mountain, and the one at 9 is a long way down: it suggests a large number of principal components given that we have only 14 variables, and the point of principal components is to replace our large number of original variables by fewer.

   Once you have an elbow, *subtract 1* to get the desired number of principal components: that is to say, 3 components for the elbow at 4 (or only 1, or 8, for the elbows at 2 and 9 respectively).

   (b) [2] Figure 27 shows the standard deviations explained by each principal component. How does this Figure support your preferred number of principal components that you obtained from the scree plot?

   **My answer:**

   Make some sort of claim that the components you are including explain noticeably more of the standard deviation than the ones you are excluding. For example, for my 3 components, the proportion of the SD explained by the first three components is "clearly" bigger than for the fourth and later components. There should be a "gap" in size of SD explained between the ones you are including and the ones you are not.

   My "clearly" is clearly (ha!) an assertion that you may take issue with.

   For this part, it doesn't matter how many components you picked in the previous part; what matters here is that you have some kind of justification for the choice you made, whatever it was.

   Extra: there is another way of choosing the number of components, usually called Kaiser's method, which is to pick as many components as have SD explained over 1. That here is again 3 components. (There happened to be a bit of a gap between the ones above 1 and the ones below, which there won't always be.) The rationale here is that everything has been standardized, so that the original variables now have SD 1, and thus if a principal component explains more than "1 variable's worth" of SD, it is worth including: as it were, it counts for more than 1 of the original variables.

   (c) [3] Figure 28 shows the loadings of all the principal components on all of the original variables. Which *three* variables are the most important in component 1? Make sure to translate the variable names into something your reader can understand.

**My answer:**

Look for the ones whose loadings are the largest *in absolute value*, that is, disregarding whether they are positive or negative. This is often tricky for component 1, because a lot of the loadings tend to be about the same size (component 1 is often a measure of "size"), but do the best you can:

- loading 0.332 for `indus`, the proportion of the tract occupied by non-retail business
- loading 0.325 for `nox`, nitrogen oxide concentration
- loading 0.324 for `tax`, property tax rate.

That's all. These three are only just the biggest, and there are many others almost as big, so it's not very helpful to think about what these have in common. (The first two are obviously related to industrialization, but the third not so much.)

(d) [2] In Figure 28, which *two* variables are the most important in component 2?

**My answer:**

This one is a bit easier to see:

- loading $-0.455$ for `medv`, median value of owner-occupied homes
- loading $-0.434$ for `rm`, average number of rooms per dwelling.

Again, that's all you need, but this time you can see that the score on component 2 will be low (negative loadings) if the tract has a lot of big, expensive houses.

(e) [4] Figure 29 is a plot of the first two component scores. Figure 30 is a five-number summary (plus the mean) of all the variables in the original dataframe. Fig 31 shows the data values for observations 163 and 164. Given this information and what you have found out so far, does it make sense that these observations appear on Figure 29 where they do? Explain briefly.

**My answer:**

First, find these two observations on the plot. They are right at the bottom, meaning that they have the lowest scores on component 2 (and are about average on component 1). From what we said about component 2 earlier, this means they should be *high* on `medv` and `rm`, because they have negative loadings on component 2.

Now use the summary and the data values to see whether they really *are* high on these variables:

- on `medv`, they are both 50, which is the highest value of all
- on `rm`, they are 7.8 and 8.38, which are not the highest but are both above the 3rd quartile, so definitely high.

This was not meant to be hard once you pulled together all the information: the plot of principal component scores (to find the observations), decide which variables play the most important role in component 2 (which you did earlier), then use the summary and data values to decide

whether the variables in question really *are* high for those observations. More careful thinking than difficult, really.

(f) [3] A biplot is shown in Figure 32. What kind of loadings should the variable `chas` (which indicates whether the census tract borders onto the Charles River) have on component 1 and component 2? Verify that it does have loadings like that.

**My answer:**

The variable `chas` points straight down on the biplot (one point).

It should have a negative loading on component 2 and a near-zero loading on component 1 (it doesn't point in the direction of component 1 at all).

Go back to Figure 28 to verify this. `chas` has a blank loading on component 1 (meaning, less than 0.1 in size), and the third-largest negative loading on component 2. This verifies that the loadings on components 1 and 2 are as we'd expect.

For each of components 1 and 2, one point for saying what you'd expect and verifying that this is indeed how it works out.

Explain your process, so that you have a completely convincing answer.

Extra 1: to revise our opinion about component 2 from before, a tract will score negatively on component 2 if:

- `medv` is high (high property values)
- `rm` is high (many rooms per house)
- `chas` is high, that is, 1 rather than zero (borders on the Charles River)

These all point to such a census tract being a prestigious place to live. (The Charles River separates the city of Boston from Cambridge, where Harvard and MIT are.)

Extra 2: having not given you a biplot in the other question, I figured I had to give you one this time, and you can use it to check your earlier work:

- `indus`, `nox`, and `tax` are among the collection of variables that point most convincingly to the right (if you can see them among the others that also do so)
- `medv` and `rm` (along with `chas`) point most convincingly downward, and there is nothing much that points upward (that would have a positive loading on component 2)
- the two observations 163 and 164 at the bottom of the plot (they might be difficult to see because I shrank the labels for the census tracts) should be high on `medv` and `rm` (and also `chas`) because they are in the direction of the pointy ends of the arrows. If you get as far as this and realize that you said earlier that they would be *low* on these variables, here is your chance to reconsider, and to go back and find a reason why they would be high rather than low. (The previous part might also have made you wonder, if that was the case.)

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(MASS)
library(car)
library(ggbiplot)
library(conflicted)
```

Figure 1: Packages

```
## Rows: 72 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: " "
## chr (2): Drug, Relapse
##
## 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.
```

```
cocaine
```

```
## # A tibble: 72 x 2
##    Drug        Relapse
##    <chr>       <chr>
##  1 Desipramine no
##  2 Lithium     yes
##  3 Placebo     no
##  4 Placebo     no
##  5 Desipramine yes
##  6 Lithium     yes
##  7 Placebo     yes
##  8 Placebo     yes
##  9 Lithium     no
## 10 Lithium     yes
## # ... with 62 more rows
```

Figure 2: Cocaine treatment data (some)

```r
cocaine.1 <- glm(factor(Relapse) ~ Drug, data = cocaine, family = "binomial")
summary(cocaine.1)
```

```
##
## Call:
## glm(formula = factor(Relapse) ~ Drug, family = "binomial", data = cocaine)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8930  -1.0383   0.6039   0.7585   1.3232
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.3365     0.4140  -0.813   0.4164
## DrugLithium   1.4351     0.6274   2.287   0.0222 *
## DrugPlacebo   1.9459     0.6866   2.834   0.0046 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 91.658  on 71  degrees of freedom
## Residual deviance: 81.220  on 69  degrees of freedom
## AIC: 87.22
##
## Number of Fisher Scoring iterations: 4
```

```r
drop1(cocaine.1, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## factor(Relapse) ~ Drug
##        Df Deviance    AIC    LRT Pr(>Chi)
## <none>      81.220 87.220
## Drug    2   91.658 93.658 10.438 0.005413 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3: Cocaine treatment model

```
Wool
```

```
##    len amp load cycles
## 1  250   8   40    674
## 2  250   8   45    370
## 3  250   8   50    292
## 4  250   9   40    338
## 5  250   9   45    266
## 6  250   9   50    210
## 7  250  10   40    170
## 8  250  10   45    118
## 9  250  10   50     90
## 10 300   8   40   1414
## 11 300   8   45   1198
## 12 300   8   50    634
## 13 300   9   40   1022
## 14 300   9   45    620
## 15 300   9   50    438
## 16 300  10   40    443
## 17 300  10   45    332
## 18 300  10   50    220
## 19 350   8   40   3636
## 20 350   8   45   3184
## 21 350   8   50   2000
## 22 350   9   40   1568
## 23 350   9   45   1070
## 24 350   9   50    566
## 25 350  10   40   1140
## 26 350  10   45    884
## 27 350  10   50    360
```

Figure 4: Wool data

```
ggplot(Wool, aes(x = len, y = cycles, fill = amp)) + geom_boxplot()
```

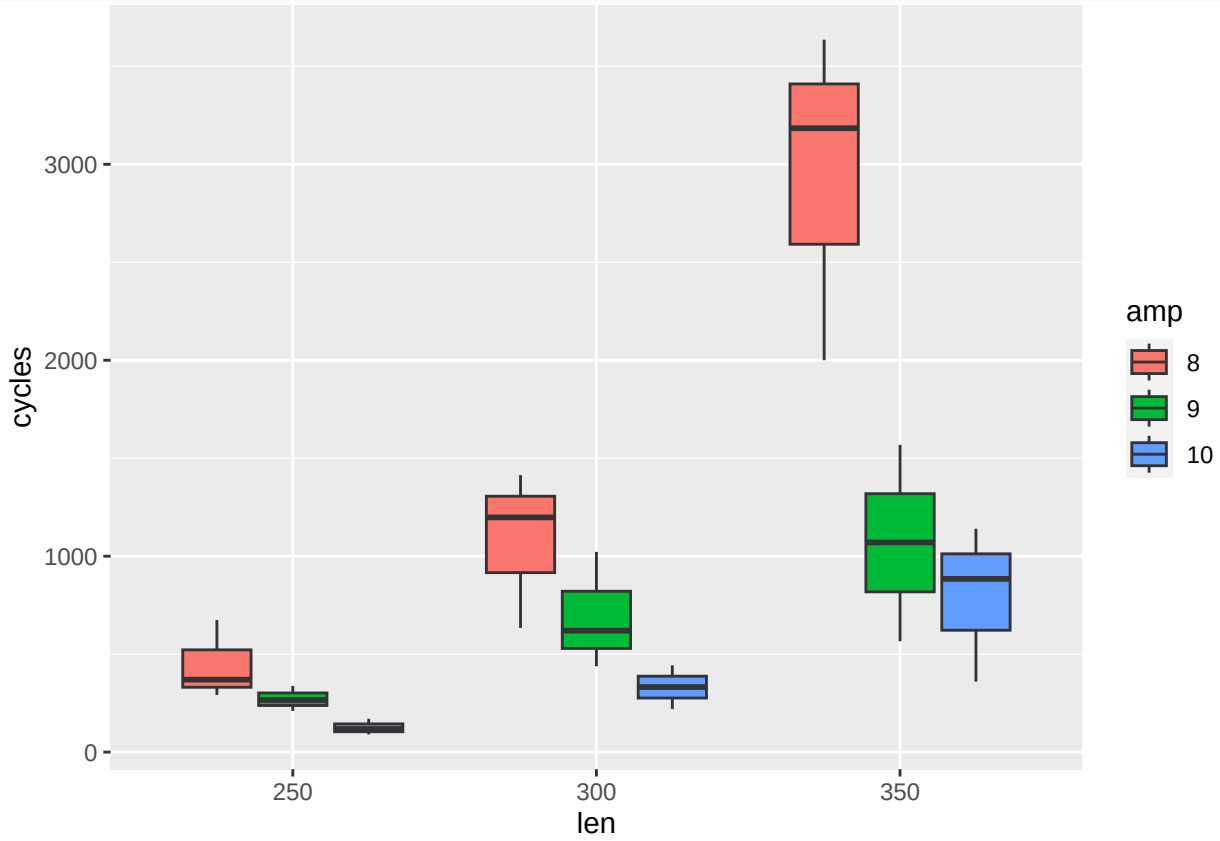



Figure 5: Wool boxplot

```
wool.1 <- aov(cycles ~ len * amp, data = Wool)
summary(wool.1)
```

```
##             Df  Sum Sq Mean Sq F value   Pr(>F)
## len          2 8182253 4091126  25.689 5.33e-06 ***
## amp          2 5624249 2812124  17.658 5.70e-05 ***
## len:amp      4 3555537  888884   5.582  0.00421 **
## Residuals   18 2866579  159254
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 6: Wool ANOVA

### SE1:

```
Wool %>% filter(len == 350) -> d1
d1.1 <- aov(cycles ~ amp, data = d1)
summary(d1.1)
```

```
##             Df  Sum Sq Mean Sq F value  Pr(>F)
## amp          2 8181550 4090775   10.93 0.00999 **
## Residuals    6 2245731  374288
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(d1.1)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = cycles ~ amp, data = d1)
##
## $amp
##            diff       lwr       upr     p adj
## 9-8  -1872.0000 -3404.681 -339.3190 0.0222658
## 10-8 -2145.3333 -3678.014 -612.6523 0.0121259
## 10-9  -273.3333 -1806.014 1259.3477 0.8516592
```

### SE2:

```
Wool %>% filter(len == 250) -> d2
d2.1 <- aov(cycles ~ amp, data = d2)
summary(d2.1)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## amp          2 153372   76686   4.947 0.0538 .
## Residuals    6  93005   15501
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(d2.1)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = cycles ~ amp, data = d2)
##
## $amp
##           diff       lwr        upr     p adj
## 9-8  -174.0000 -485.9083 137.908258 0.2761411
## 10-8 -319.3333 -631.2416  -7.425075 0.0457129
## 10-9 -145.3333 -457.2416 166.574925 0.3854304
```

Figure 7: Wool simple effects

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

```
##    Gender Freshwater Marine   Origin      combo
## 1       1        119    474  Alaskan  Alaskan-1
## 2       1        144    403 Canadian Canadian-1
## 3       1         94    440  Alaskan  Alaskan-1
## 4       1         85    451  Alaskan  Alaskan-1
## 5       2        120    369 Canadian Canadian-2
## 6       2         92    404  Alaskan  Alaskan-2
## 7       2        124    389 Canadian Canadian-2
## 8       2        145    355 Canadian Canadian-2
## 9       1         99    481  Alaskan  Alaskan-1
## 10      2        101    474  Alaskan  Alaskan-2
## 11      2        123    349 Canadian Canadian-2
## 12      2        150    354 Canadian Canadian-2
## 13      1        154    390 Canadian Canadian-1
## 14      1         70    397  Alaskan  Alaskan-1
## 15      2        149    393 Canadian Canadian-2
## 16      1        148    371 Canadian Canadian-1
## 17      2         91    469  Alaskan  Alaskan-2
## 18      1        148    383 Canadian Canadian-1
## 19      1        156    419 Canadian Canadian-1
## 20      1        152    301 Canadian Canadian-1
```

Figure 8: Salmon data (20 randomly chosen rows)

```
##      Freshwater Marine
## [1,]        108    368
## [2,]        131    355
## [3,]        105    469
## [4,]         86    506
## [5,]         99    402
## [6,]         87    423
```

```
salmon.1 <- manova(response ~ Gender*Origin, data = salmon)
summary(salmon.1)
```

```
##               Df  Pillai approx F num Df den Df Pr(>F)
## Gender         1 0.00325    0.155      2     95 0.8568
## Origin         1 0.67939  100.657      2     95 <2e-16 ***
## Gender:Origin  1 0.03273    1.607      2     95 0.2059
## Residuals     96
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
salmon.2 <- manova(response ~ Gender + Origin, data = salmon)
summary(salmon.2)
```

```
##           Df  Pillai approx F num Df den Df Pr(>F)
## Gender     1 0.00320    0.154      2     96 0.8572
## Origin     1 0.67937  101.703      2     96 <2e-16 ***
## Residuals 97
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
salmon.3 <- manova(response ~ Origin, data = salmon)
summary(salmon.3)
```

```
##           Df Pillai approx F num Df den Df    Pr(>F)
## Origin     1  0.679   102.59      2     97 < 2.2e-16 ***
## Residuals 98
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 9: Salmon MANOVA

```
salmon.4 <- lda(combo ~ Freshwater + Marine, data = salmon)
salmon.4
```

```
## Call:
## lda(combo ~ Freshwater + Marine, data = salmon)
##
## Prior probabilities of groups:
##  Alaskan-1  Alaskan-2 Canadian-1 Canadian-2
##       0.26       0.24       0.26       0.24
##
## Group means:
##            Freshwater   Marine
## Alaskan-1    96.57692 423.6538
## Alaskan-2   100.33333 436.1667
## Canadian-1  139.53846 369.0000
## Canadian-2  135.20833 364.0417
##
## Coefficients of linear discriminants:
##                    LD1         LD2
## Freshwater  0.04419519 -0.03805305
## Marine     -0.01785288 -0.02360065
##
## Proportion of trace:
##    LD1    LD2
## 0.9836 0.0164
```

Figure 10: Salmon discriminant analysis

```
p <- predict(salmon.4)
d <- cbind(salmon, p)
```

Figure 11: Salmon discriminant analysis part 2

Note: the `cbbPalette` and the `scale_colour_manual` draw the points with (I am told) colour-blind-friendly colours. If it is still impossible for you to distinguish the colours, ask an invigilator for help identifying the colour of some points.

```
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
ggplot(d, aes(x = x.LD1, y = x.LD2, colour = combo)) +
  geom_point() + scale_colour_manual(values = cbbPalette)
```

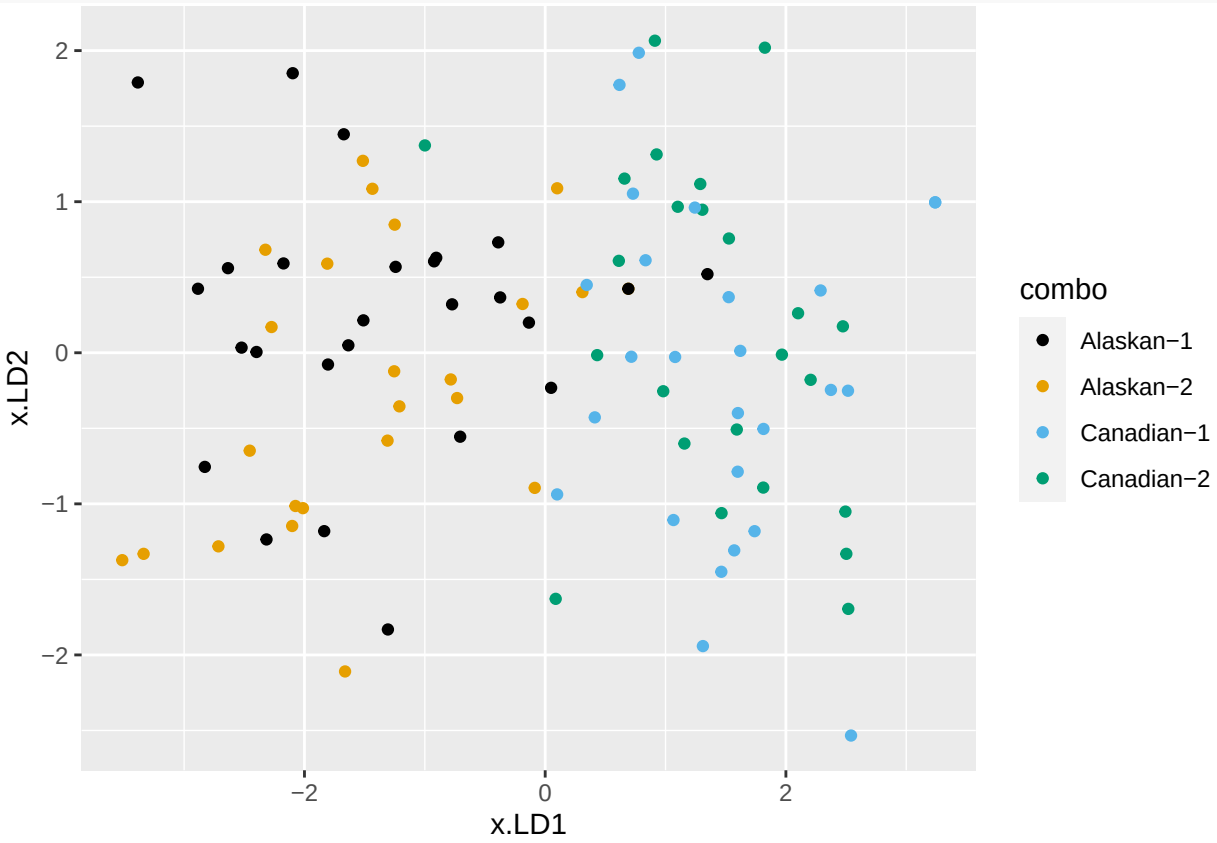



Figure 12: Salmon discriminant scores plot

```
with(d, table(combo, class))
```

```
##             class
## combo        Alaskan-1 Alaskan-2 Canadian-1 Canadian-2
##   Alaskan-1         18         5          1          2
##   Alaskan-2          9        12          0          3
##   Canadian-1         0         0         17          9
##   Canadian-2         1         0         13         10
```

Figure 13: Salmon discriminant misclassification table

```
se
```

```
##    subject treatment se1 se2 se3
## 1        1   Control  14  13  15
## 2        2   Control  13  14  17
## 3        3   Control  17  12  16
## 4        4   Control  11  11  12
## 5        5   Control  16  15  14
## 6        6   Control  17  18  18
## 7        7   Control  17  16  19
## 8        8   Control  13  15  15
## 9        9   Control  14  14  15
## 10      10   Control  14  15  13
## 11      11   Control  16  16  11
## 12      12   Control  15  13  16
## 13      13      Diet  12  11  14
## 14      14      Diet  13  14  15
## 15      15      Diet  17  11  18
## 16      16      Diet  16  15  18
## 17      17      Diet  16  17  15
## 18      18      Diet  13  11  18
## 19      19      Diet  12  11  14
## 20      20      Diet  12  11  11
## 21      21      Diet  17  16  19
## 22      22      Diet  19  19  19
## 23      23      Diet  15  15  15
## 24      24      Diet  16  14  18
## 25      25    DietEx  15  11  19
## 26      26    DietEx  16  12  18
## 27      27    DietEx  13  12  17
## 28      28    DietEx  16  13  17
## 29      29    DietEx  13  13  16
## 30      30    DietEx  15  12  18
## 31      31    DietEx  15  13  18
## 32      32    DietEx  16  14  17
## 33      33    DietEx  16  16  19
## 34      34    DietEx  17  17  17
```

Figure 14: Self esteem data

```
se %>%
  pivot_longer(starts_with("se"), names_to = "time", values_to = "self_esteem") -> xx
```

Figure 15: Self esteem interaction plot part 1

```
xx %>%
  group_by(treatment, time) %>%
  summarize(mean_se = mean(self_esteem)) %>%
  ggplot(aes(x = time, y = mean_se, colour = treatment, group = treatment)) +
  geom_point() + geom_line()
```

```
## `summarise()` has grouped output by 'treatment'. You can override using the `.groups`
## argument.
```

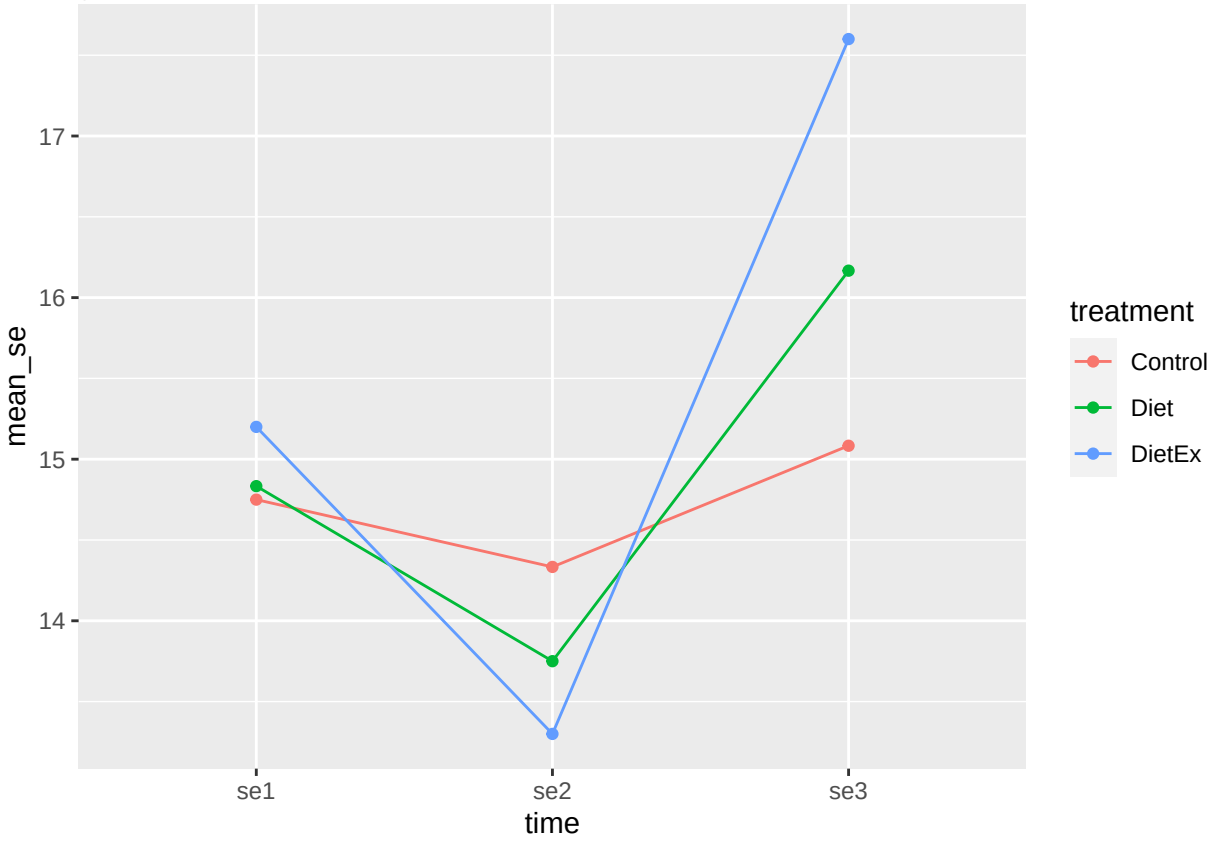



Figure 16: Self esteem interaction plot part 2

```
WeightLoss %>%
  select(starts_with("se")) %>%
  as.matrix() -> response
se.1 <- lm(response ~ group, data = WeightLoss)
times <- colnames(response)
times.df <- data.frame(times = factor(times))
se.2 <- Manova(se.1, idata = times.df, idesign = ~times)
summary(se.2)$univariate.tests
```

```
##              Sum Sq num Df Error SS den Df   F value    Pr(>F)
## (Intercept) 22890.0      1   278.94     31 2543.8949 < 2.2e-16 ***
## group           7.0      2   278.94     31    0.3902  0.680205
## times          96.7      2   134.58     62   22.2807 5.111e-08 ***
## group:times    34.7      4   134.58     62    3.9962  0.006003 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(se.2)$sphericity.tests # Mauchly's test
```

```
##             Test statistic  p-value
## times              0.75295 0.014173
## group:times        0.75295 0.014173
```

```
summary(se.2)$pval.adjustments
```

```
##               GG eps   Pr(>F[GG])    HF eps   Pr(>F[HF])
## times       0.801891 7.595864e-07 0.8389008 4.583464e-07
## group:times 0.801891 1.105415e-02 0.8389008 9.855788e-03
## attr(,"na.action")
## (Intercept)       group
##           1           2
## attr(,"class")
## [1] "omit"
```

Figure 17: Self esteem ANOVA: univariate tests, Mauchly's test, Greenhouse-Geisser and Huynh-Feldt adjustments

- `BR` brakes
- `FU` fuel system
- `EL` electrical
- EX exhaust
- `ST` steering
- `EM` engine, mechanical
- `RS` rattles and squeaks
- `RA` rear axle
- `RU` rust
- `SA` shock absorbers
- `TC` transmission or clutch
- `WA` wheel alignment
- `OT` other

Figure 18: Cars data column names

```
us.car.repair.1969
```

```
##               model BR FU EL EX ST EM RS RA RU SA TC WA OT
## 1  AMC Ambassador 8  +  -  -  -  -  -  -  +  -  -  -  -  -
## 2   Buick Special 6  -  -  -  -  -  -  +  -  +  -  -  -  +
## 3   Buick Special 8  -  -  -  -  -  -  +  -  -  +  -  +  +
## 4      Buick 8 Full  -  -  -  +  -  +  +  -  +  +  -  +  -
## 5     Buick Riviera  -  -  +  +  -  -  -  -  -  +  -  -  -
## 6          Chevy II  -  +  -  -  +  -  +  +  +  -  -  +  -
## 7        Chevelle 6  -  -  -  -  -  +  +  -  +  -  -  -  -
## 8        Chevelle 8  -  +  -  +  +  -  +  -  +  +  -  +  -
## 9    Chevrolet Full  -  +  +  +  +  -  +  +  +  +  +  +  -
## 10        Corvair 6  -  +  -  -  +  +  -  +  -  +  +  +  +
## 11         Corvette  -  -  -  +  -  -  +  +  -  -  +  -  -
## 12 Chrysler Newport  +  -  -  -  -  -  -  -  -  -  -  -  -
## 13       New Yorker  +  -  -  -  -  -  -  +  -  -  -  -  +
## 14  Dodge Full Size  +  -  -  -  -  -  +  -  -  +  -  -  -
## 15         Falcon 6  -  -  -  -  -  -  +  -  -  -  +  +  -
## 16       Fairlane 6  -  -  -  -  -  -  +  -  -  -  -  -  -
## 17       Fairlane 8  -  -  -  +  -  -  +  +  -  -  -  +  -
## 18   Ford Full Size  -  -  -  +  +  -  -  -  -  +  -  +  +
## 19      Thunderbird  -  -  +  -  +  +  -  -  -  -  -  +  +
## 20     Mercury Full  -  -  -  -  -  -  -  -  -  -  -  +  -
## 21        Olds Full  +  +  -  -  -  -  +  -  -  +  -  +  -
## 22    Plymouth Full  +  -  -  -  -  -  -  -  -  -  -  -  -
## 23  Pontiac Tempest  -  +  -  -  -  -  +  -  +  +  -  +  -
## 24     Pontiac Full  +  +  +  -  -  -  +  -  +  +  -  +  -
## 25  Rambler Rebel 6  -  -  +  -  -  -  -  +  -  -  +  -  +
## 26         Mercedes  -  -  -  -  -  -  -  -  -  +  -  -  -
## 27          MG 1100  -  -  +  +  -  -  -  -  -  -  -  -  -
## 28          Peugeot  -  -  -  -  -  -  -  -  -  -  +  -  -
## 29          Porsche  -  -  -  -  -  -  -  -  -  -  -  -  -
## 30          Renault  -  -  -  -  -  -  -  -  -  -  +  -  -
## 31            Volvo  -  -  -  +  -  -  -  +  -  -  -  -  -
## 32           VW bug  +  -  +  +  +  +  -  -  -  -  +  -  -
## 33           VW bus  -  -  +  -  -  +  -  -  +  -  +  -  -
```

Figure 19: Cars data

```
cars.1 <- hclust(dissim, method = "ward.D")
plot(cars.1)
rect.hclust(cars.1, 2)
```

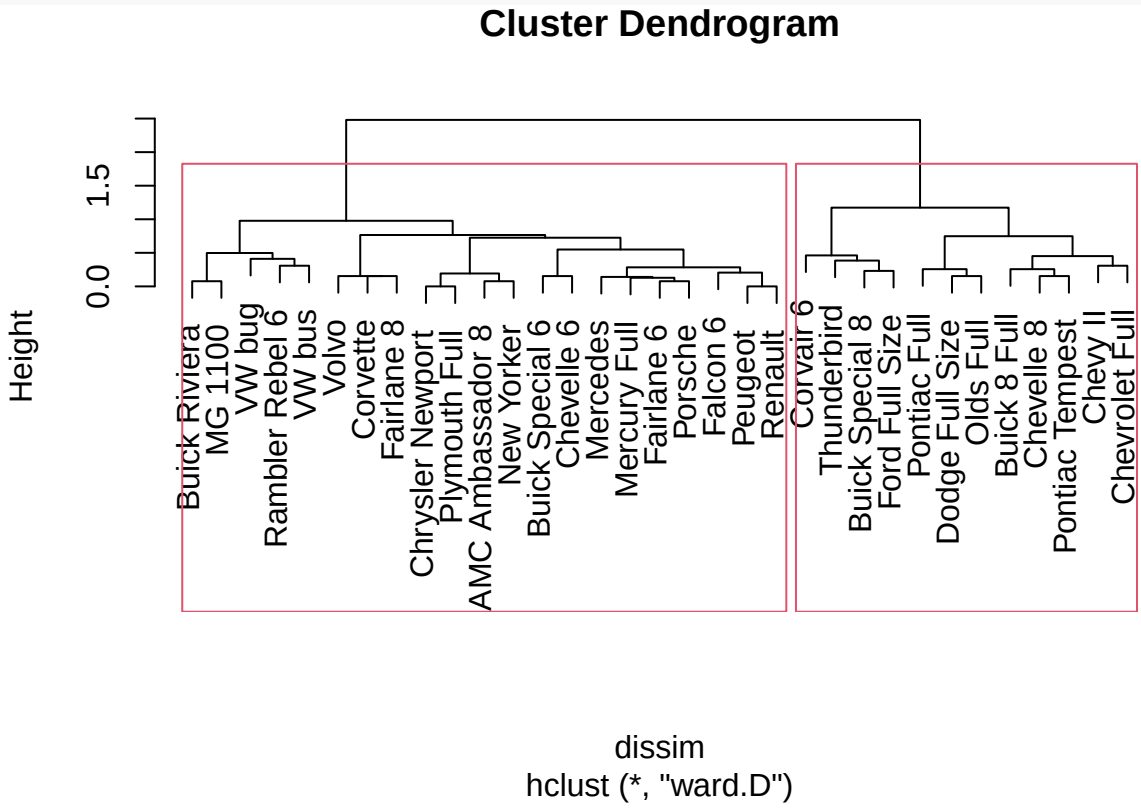



Figure 20: Cars cluster analysis

```
us.car.repair.1969 %>% pivot_longer(-model, names_to = "repair", values_to = "compare_avg") %>%
  group_by(model) %>%
  count(compare_avg) %>%
  filter(compare_avg == "+") %>%
  select(-compare_avg) -> problems
problems
```

```
## # A tibble: 32 x 2
## # Groups:   model [32]
##    model                n
##    <chr>            <int>
##  1 AMC Ambassador 8     2
##  2 Buick 8 Full         6
##  3 Buick Riviera        3
##  4 Buick Special 6      3
##  5 Buick Special 8      4
##  6 Chevelle 6           3
##  7 Chevelle 8           7
##  8 Chevrolet Full      10
##  9 Chevy II             6
## 10 Chrysler Newport     1
## # ... with 22 more rows
```

Figure 21: Cars: total number of above-average items by model

```
cutree(cars.1, 2) %>% enframe(name = "model", value = "cluster") -> clusters
clusters
```

```
## # A tibble: 33 x 2
##    model            cluster
##    <chr>              <int>
##  1 AMC Ambassador 8       1
##  2 Buick Special 6        1
##  3 Buick Special 8        2
##  4 Buick 8 Full           2
##  5 Buick Riviera          1
##  6 Chevy II               2
##  7 Chevelle 6             1
##  8 Chevelle 8             2
##  9 Chevrolet Full         2
## 10 Corvair 6              2
## # ... with 23 more rows
```

Figure 22: Cars: cluster membership

```
problems %>% left_join(clusters) %>%
  mutate(cluster = factor(cluster)) %>%
  ggplot(aes(x = cluster, y = n)) + geom_boxplot()
```

```
## Joining with `by = join_by(model)`
```

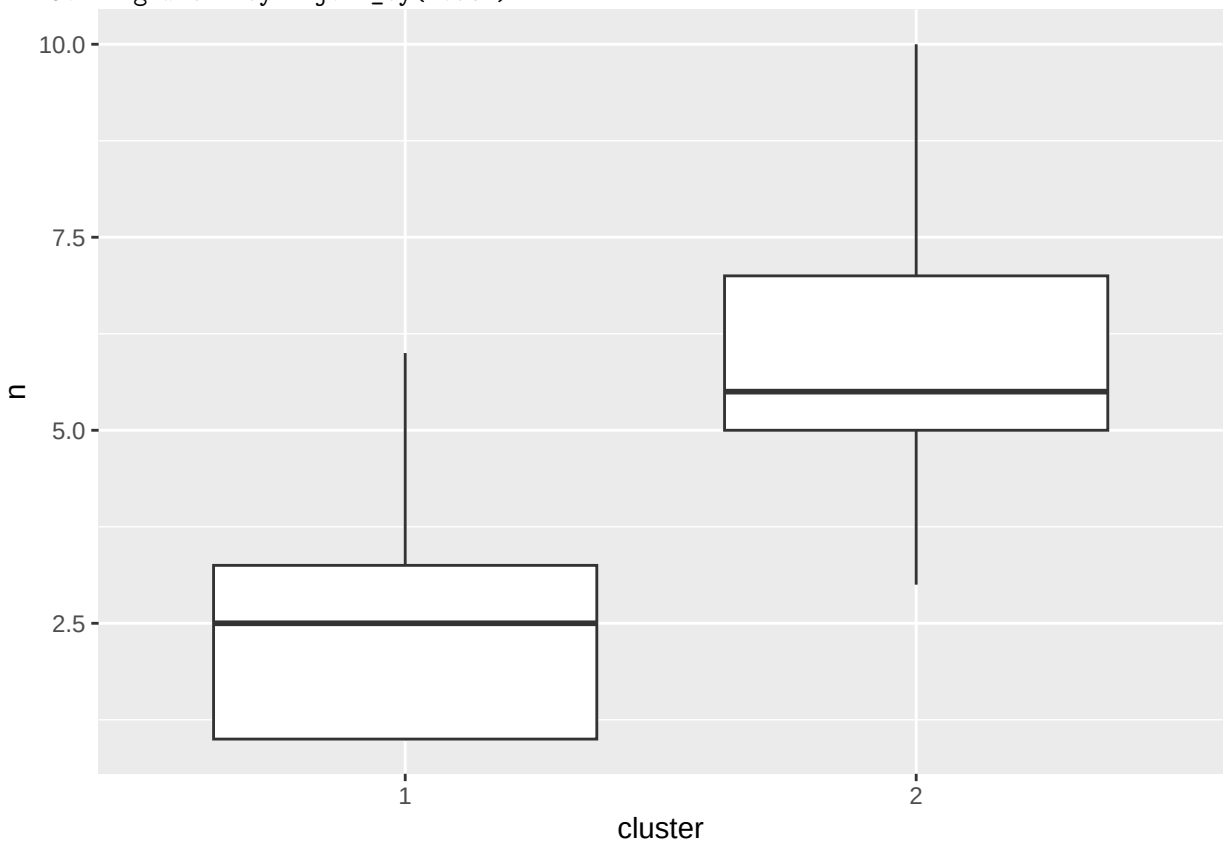



Figure 23: Cars: more computation and boxplot

- `crim` per capita crime rate.
- `zn` proportion of residential land zoned for lots over 25,000 sq.ft.
- `indus` proportion of tract occupied by non-retail business (ie. businesses that are not stores).
- `chas` Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
- `nox` nitrogen oxides concentration (parts per 10 million).
- `rm` average number of rooms per dwelling.
- `age` proportion of owner-occupied units built prior to 1940.
- `dis` weighted mean of distances to five Boston employment centres.
- `rad` index of accessibility to radial highways.
- `tax` full-value property-tax rate per $10,000.
- `ptratio` pupil-teacher ratio for schools in that tract.
- `black` A formula that is close to zero if the tract has proportion of Black people close to average (for Boston), and is high if that proportion is much higher *or lower* than average.
- `lstat` percent of the population that is of lower socio-economic status.
- `medv` median value of owner-occupied homes in $1000s.

Figure 24: Variables measured on Boston census tracts

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

```
## # A tibble: 20 x 14
##       crim    zn indus  chas   nox    rm   age   dis   rad   tax ptratio black lstat  medv
##      <dbl> <dbl> <dbl> <int> <dbl> <dbl> <dbl> <dbl> <int> <dbl>   <dbl> <dbl> <dbl> <dbl>
##  1 0.00632  18    2.31     0 0.538  6.58  65.2  4.09     1   296    15.3  397.  4.98  24
##  2 0.0273    0    7.07     0 0.469  6.42  78.9  4.97     2   242    17.8  397.  9.14  21.6
##  3 0.0273    0    7.07     0 0.469  7.18  61.1  4.97     2   242    17.8  393.  4.03  34.7
##  4 0.0324    0    2.18     0 0.458  7.00  45.8  6.06     3   222    18.7  395.  2.94  33.4
##  5 0.0690    0    2.18     0 0.458  7.15  54.2  6.06     3   222    18.7  397.  5.33  36.2
##  6 0.0298    0    2.18     0 0.458  6.43  58.7  6.06     3   222    18.7  394.  5.21  28.7
##  7 0.0883   12.5  7.87     0 0.524  6.01  66.6  5.56     5   311    15.2  396. 12.4   22.9
##  8 0.145    12.5  7.87     0 0.524  6.17  96.1  5.95     5   311    15.2  397. 19.2   27.1
##  9 0.211    12.5  7.87     0 0.524  5.63 100    6.08     5   311    15.2  387. 29.9   16.5
## 10 0.170    12.5  7.87     0 0.524  6.00  85.9  6.59     5   311    15.2  387. 17.1   18.9
## 11 0.225    12.5  7.87     0 0.524  6.38  94.3  6.35     5   311    15.2  393. 20.4   15
## 12 0.117    12.5  7.87     0 0.524  6.01  82.9  6.23     5   311    15.2  397. 13.3   18.9
## 13 0.0938   12.5  7.87     0 0.524  5.89  39    5.45     5   311    15.2  390. 15.7   21.7
## 14 0.630     0    8.14     0 0.538  5.95  61.8  4.71     4   307    21    397.  8.26  20.4
## 15 0.638     0    8.14     0 0.538  6.10  84.5  4.46     4   307    21    380. 10.3   18.2
## 16 0.627     0    8.14     0 0.538  5.83  56.5  4.50     4   307    21    396.  8.47  19.9
## 17 1.05      0    8.14     0 0.538  5.94  29.3  4.50     4   307    21    387.  6.58  23.1
## 18 0.784     0    8.14     0 0.538  5.99  81.7  4.26     4   307    21    387. 14.7   17.5
## 19 0.803     0    8.14     0 0.538  5.46  36.6  3.80     4   307    21    289. 11.7   20.2
## 20 0.726     0    8.14     0 0.538  5.73  69.5  3.80     4   307    21    391. 11.3   18.2
```

Figure 25: Boston census tract data (some)

```r
boston.1 <- princomp(Boston, cor = TRUE)
ggscreeplot(boston.1)
```

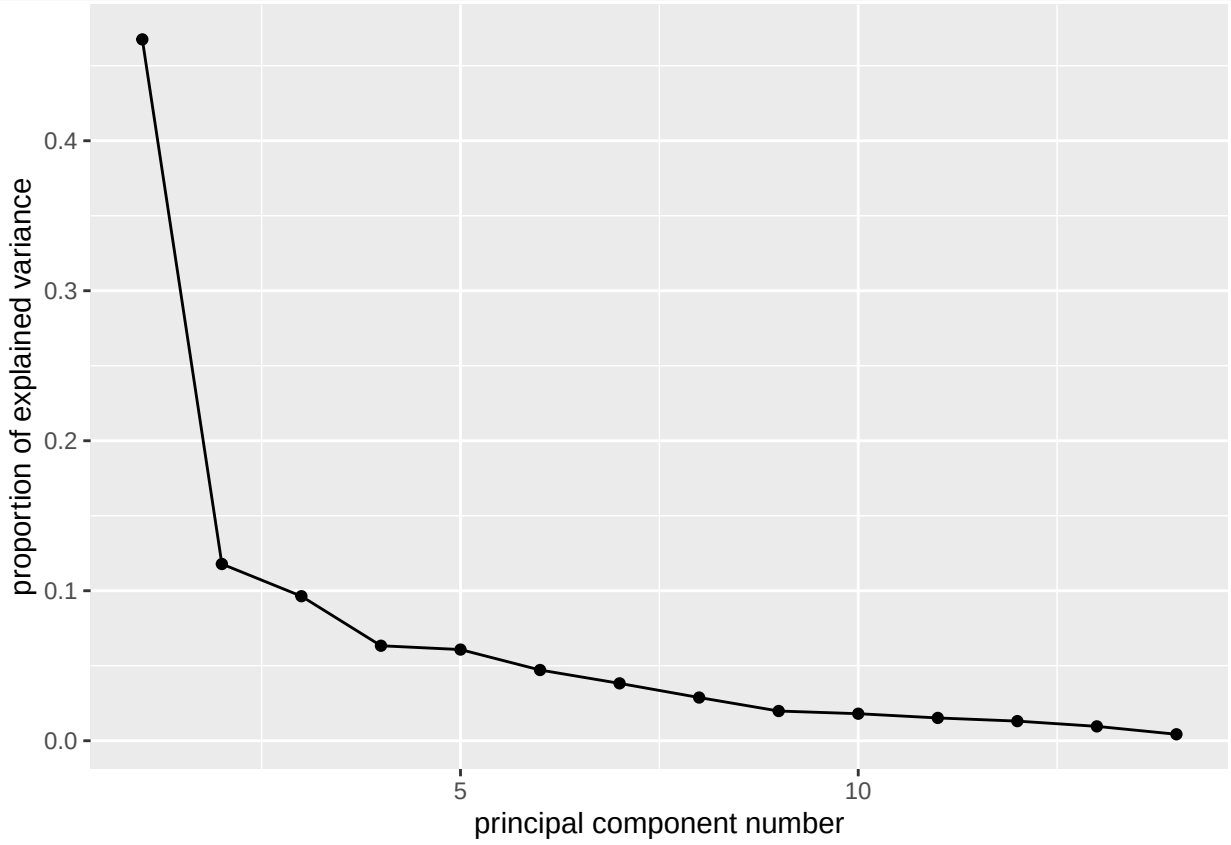



Figure 26: Boston scree plot

```r
boston.1
```

```
## Call:
## princomp(x = Boston, cor = TRUE)
##
## Standard deviations:
##    Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7    Comp.8    Comp.9
## 2.5585132 1.2843410 1.1614241 0.9415625 0.9224421 0.8124105 0.7317177 0.6348831 0.5265582
##   Comp.10   Comp.11   Comp.12   Comp.13   Comp.14
## 0.5022524 0.4612919 0.4277704 0.3660733 0.2456149
##
##  14  variables and  506 observations.
```

Figure 27: Boston SD explained

```
boston.1$loadings
```

```
##
## Loadings:
##         Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10 Comp.11
## crim     0.242         0.395  0.100         0.225  0.777  0.157  0.254
## zn      -0.245  0.148  0.395  0.343  0.114  0.336 -0.274 -0.380  0.383 -0.246   0.128
## indus    0.332 -0.127                               -0.340  0.172  0.627  0.255  -0.274
## chas           -0.411 -0.125  0.700 -0.535 -0.163
## nox      0.325 -0.254                0.195  0.149 -0.198                0.212   0.437
## rm      -0.203 -0.434  0.353 -0.293        -0.131        -0.438         0.526  -0.224
## age      0.297 -0.260 -0.201         0.150         0.119 -0.588        -0.246   0.330
## dis     -0.298  0.359  0.157  0.185 -0.106        -0.104 -0.128 -0.176  0.299   0.115
## rad      0.303         0.419        -0.230  0.135 -0.137        -0.463 -0.116
## tax      0.324         0.343        -0.163  0.188 -0.314        -0.179
## ptratio  0.208  0.315        -0.342 -0.616 -0.279        -0.283  0.275 -0.160
## black   -0.197        -0.361 -0.202 -0.367  0.786                       0.146
## lstat    0.311  0.201 -0.161  0.243  0.178               -0.357 -0.172         -0.683
## medv    -0.267 -0.445  0.163 -0.180                      0.152        -0.576  -0.242
##         Comp.12 Comp.13 Comp.14
## crim
## zn      -0.221  -0.132
## indus    0.348          -0.235
## chas
## nox     -0.449   0.525
## rm      -0.126
## age      0.486
## dis      0.494   0.552
## rad                     -0.635
## tax      0.170  -0.243   0.699
## ptratio -0.232   0.188
## black
## lstat   -0.182   0.249
## medv             0.470   0.134
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10
## SS loadings     1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000   1.000
## Proportion Var  0.071  0.071  0.071  0.071  0.071  0.071  0.071  0.071  0.071   0.071
## Cumulative Var  0.071  0.143  0.214  0.286  0.357  0.429  0.500  0.571  0.643   0.714
##                Comp.11 Comp.12 Comp.13 Comp.14
## SS loadings      1.000   1.000   1.000   1.000
## Proportion Var   0.071   0.071   0.071   0.071
## Cumulative Var   0.786   0.857   0.929   1.000
```

Figure 28: Boston principal component loadings

```
boston.1$scores %>% as_tibble() %>%
  mutate(r = row_number()) %>%
  ggplot(aes(x = Comp.1, y = Comp.2, label = r)) + geom_text()
```

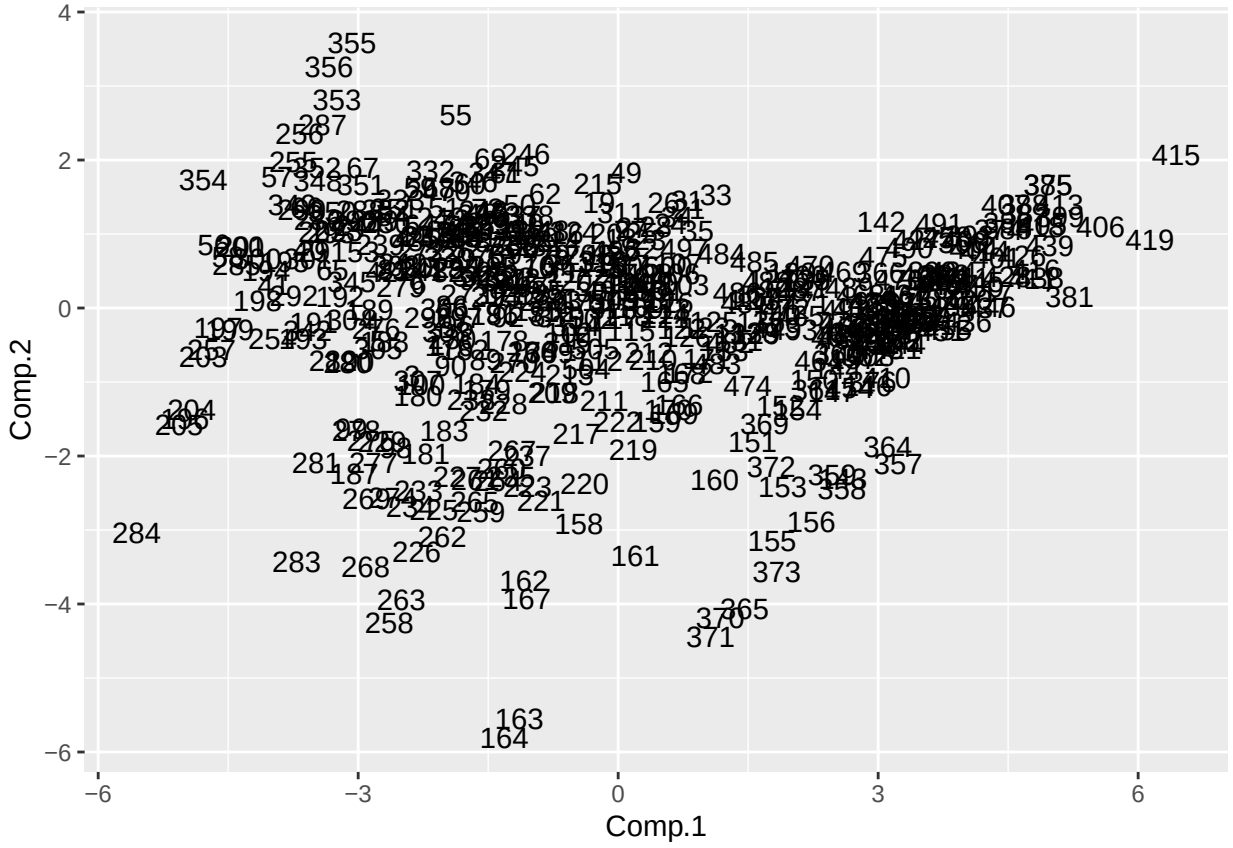



Figure 29: Boston principal component scores plot

```
summary(Boston)
```

```
##       crim                zn             indus            chas              nox        
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000   Min.   :0.3850  
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000   1st Qu.:0.4490  
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000   Median :0.5380  
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917   Mean   :0.5547  
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000   3rd Qu.:0.6240  
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000   Max.   :0.8710  
##        rm             age              dis              rad              tax        
##  Min.   :3.561   Min.   :  2.90   Min.   : 1.130   Min.   : 1.000   Min.   :187.0  
##  1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100   1st Qu.: 4.000   1st Qu.:279.0  
##  Median :6.208   Median : 77.50   Median : 3.207   Median : 5.000   Median :330.0  
##  Mean   :6.285   Mean   : 68.57   Mean   : 3.795   Mean   : 9.549   Mean   :408.2  
##  3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188   3rd Qu.:24.000   3rd Qu.:666.0  
##  Max.   :8.780   Max.   :100.00   Max.   :12.127   Max.   :24.000   Max.   :711.0  
##     ptratio          black            lstat            medv      
##  Min.   :12.60   Min.   :  0.32   Min.   : 1.73   Min.   : 5.00  
##  1st Qu.:17.40   1st Qu.:375.38   1st Qu.: 6.95   1st Qu.:17.02  
##  Median :19.05   Median :391.44   Median :11.36   Median :21.20  
##  Mean   :18.46   Mean   :356.67   Mean   :12.65   Mean   :22.53  
##  3rd Qu.:20.20   3rd Qu.:396.23   3rd Qu.:16.95   3rd Qu.:25.00  
##  Max.   :22.00   Max.   :396.90   Max.   :37.97   Max.   :50.00
```

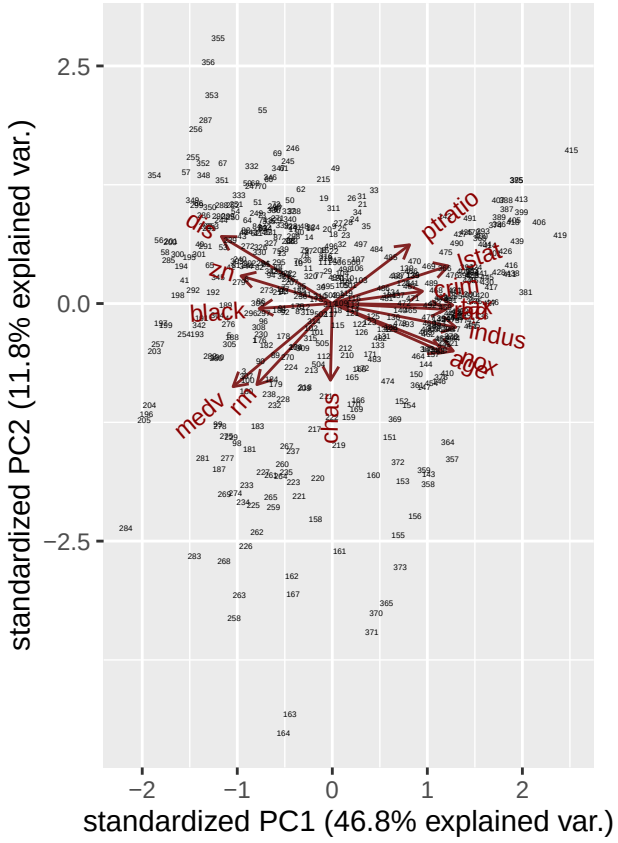
Figure 30: Boston summary

```
Boston %>% slice(163, 164)
```

```
## # A tibble: 2 x 14
##    crim    zn indus  chas   nox    rm   age   dis   rad   tax ptratio black lstat  medv
##   <dbl> <dbl> <dbl> <int> <dbl> <dbl> <dbl> <dbl> <int> <dbl>   <dbl> <dbl> <dbl> <dbl>
## 1  1.83     0  19.6     1 0.605  7.80  98.2  2.04     5   403    14.7  390.  1.92    50
## 2  1.52     0  19.6     1 0.605  8.38  93.9  2.16     5   403    14.7  388.  3.32    50
```

Figure 31: Boston data values for observations 163 and 164

```
ggbiplot(boston.1, labels = 1:506, labels.size = 1, alpha = 0.5)
```




I tried to make this come out bigger, but it wouldn't.

Figure 32: Boston data biplot