# University of Toronto Scarborough
# Department of Computer and Mathematical Sciences
# STAC32 (K. Butler), Midterm Exam
# October 21, 2017

READ THE BOX BELOW, AND FOLLOW THE INSTRUCTIONS IN IT.

Aids allowed:
- My lecture slides
- Any notes that you have taken in this course
- Your assignments and feedback on them
- My assignment solutions
- The course R text
- The course SAS text
- Non-programmable, non-communicating calculator

Past exams are *not* allowed.

Before you begin, complete the signature sheet, but sign it only when the invigilator collects it. The signature sheet shows that you were present at the exam.

This exam has 51 numbered pages of questions. Please check to see that you have all the pages.

In addition, you should have an additional booklet of output to refer to during the exam. Contact an invigilator if you do not have this.

Answer each question in the space provided (under the question). If you need more space, use the backs of the pages, but be sure to draw the marker's attention to where the rest of the answer may be found.

The maximum marks available for each part of each question are shown next to the question part. In addition, the total marks available for each *page* are shown at the bottom of the page, and also in the table on the next page.

When giving SAS code, you can provide code that runs either on the online version of SAS Studio, or on the version that runs on a virtual machine. Either version is acceptable.

Code for R graphs should be in `ggplot` style, as in lecture. There may be partial credit for "base" graphs.

For any questions below involving R code, you may assume that this code has already been run:

```
library(tidyverse)
```

*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.*

Last name: _____

First name: _____

Student number: _____

For marker's use only:

| Page | Points | Score |
|---|---|---|
| 1 | 3 | |
| 2 | 2 | |
| 5 | 4 | |
| 6 | 5 | |
| 9 | 3 | |
| 12 | 3 | |
| 15 | 3 | |
| 21 | 4 | |
| 23 | 2 | |
| 24 | 2 | |
| 26 | 3 | |
| 28 | 3 | |
| 30 | 2 | |
| 31 | 2 | |
| 32 | 3 | |
| 35 | 3 | |
| 36 | 2 | |
| 37 | 3 | |
| 38 | 3 | |
| 41 | 2 | |
| 42 | 2 | |
| 43 | 2 | |
| 44 | 3 | |
| 46 | 5 | |
| 47 | 3 | |

1. A computer science student wants to know whether it takes longer to download files at certain times of the day. To assess this, she placed a file on a remote server, and repeatedly downloaded it from there at different times of day: "early" (7:00am), "evening" (5:00pm) and "late-night" (midnight). The student downloaded the file 16 times at each of the three times of day, recording the download time in seconds, for a total of 48 observations. The data are shown in Figure 1 of the booklet of code and output.

   (a) (3 marks) Is the data file formatted as a CSV, space-delimited, or something else? Make a reasonable assumption about the format of the data, and give code to read the data file into R. The data is in the file `download.txt` in the current folder (the folder in which R is working). Use the name `downloads` for your input data frame.

**Solution:** CSV means "comma-separated values", so it is not that. For space-delimited data, the values would have to be separated by exactly one space, which they are not. (Don't get taken in by the file having an extension `.txt`: what matters is what it *contains*, which is why I showed it to you.) The column of download times often starts in the same physical column of the data file, and when the time of day is long enough, the download time is further to the right. This suggests that the values might be separated by tabs (which is actually the case). Even if you are not sure about this, it seems a reasonable assumption to make. Another plausible possibility is `read_table`, but the columns need to be aligned all the way down, which they are not. I gave that two points.

With that in mind, `read_tsv` is the thing, just as for the Australian athletes. You only need to give the filename, since it's in the current folder (or run `file.choose` if you prefer):

```
downloads=read_tsv("download.txt")
```

```
## Parsed with column specification:
## cols(
##   time_of_day = col_character(),
##   download_time = col_integer()
## )
```

```
downloads
```

```
## # A tibble: 48 x 2
##    time_of_day download_time
##          <chr>         <int>
##  1       early            69
##  2       early           138
##  3       early            75
##  4       early           186
##  5       early            68
##  6       early           217
##  7       early            93
##  8       early            90
##  9       early            71
## 10       early           154
## # ... with 38 more rows
```

That works, and is really the best way to do it.

If you made a different and less reasonable assumption about the data format and coded it correctly, expect to get one point.

If you can figure out how "tab" looks in R, that will also work:

```
downloads=read_delim("download.txt","\t")

## Parsed with column specification:
## cols(
##   time_of_day = col_character(),
##   download_time = col_integer()
## )
```

There is an old-fashioned command `read.table` with a dot rather than an underscore, that was pretty good at reading in stuff like this:

```
downloads2=read.table("download.txt",header=T)
head(downloads2)

##   time_of_day download_time
## 1       early            69
## 2       early           138
## 3       early            75
## 4       early           186
## 5       early            68
## 6       early           217
```

In the (rather unlikely) event that you tried that and coded it properly, that will do (but see below). What `read.table` does is to read data values separated by one or more whitespace characters, by which I mean spaces or tabs. (Normally "newline" is included among the whitespace characters, but not here.) Even if those tabs had been more than one space, this would have read the data in successfully.

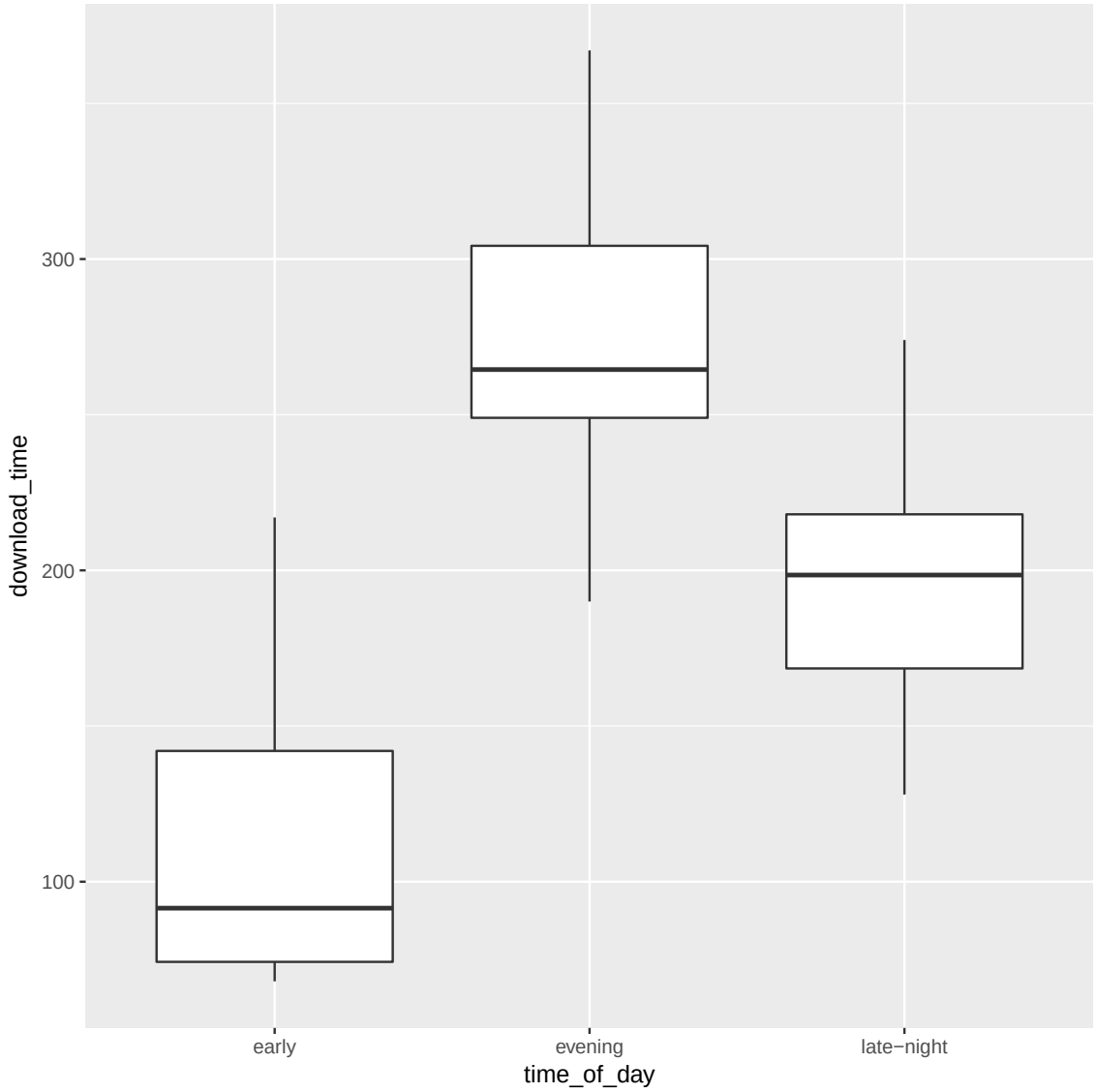
The downside of this is that you get an old-fashioned data frame rather than a "tibble", so if you type its name you get all its rows, rather than just the first ten. `head`, if you have not run into it before, displays the first six of anything, in this case rows of the data frame.

This isn't something we've done in class this year, so if you want to use it, you need (if you want full marks) to offer some kind of justification for why it would work, such as "the data values are separated by one or more spaces". What looks like multiple spaces are in fact tabs, but `read.table` works both ways, and Figure 1 could be read either way, so I'm good with that.

(b) (2 marks) Give R code to draw a boxplot of download time for each time of day. (You may assume that you successfully read the data into a data frame called `downloads`.)

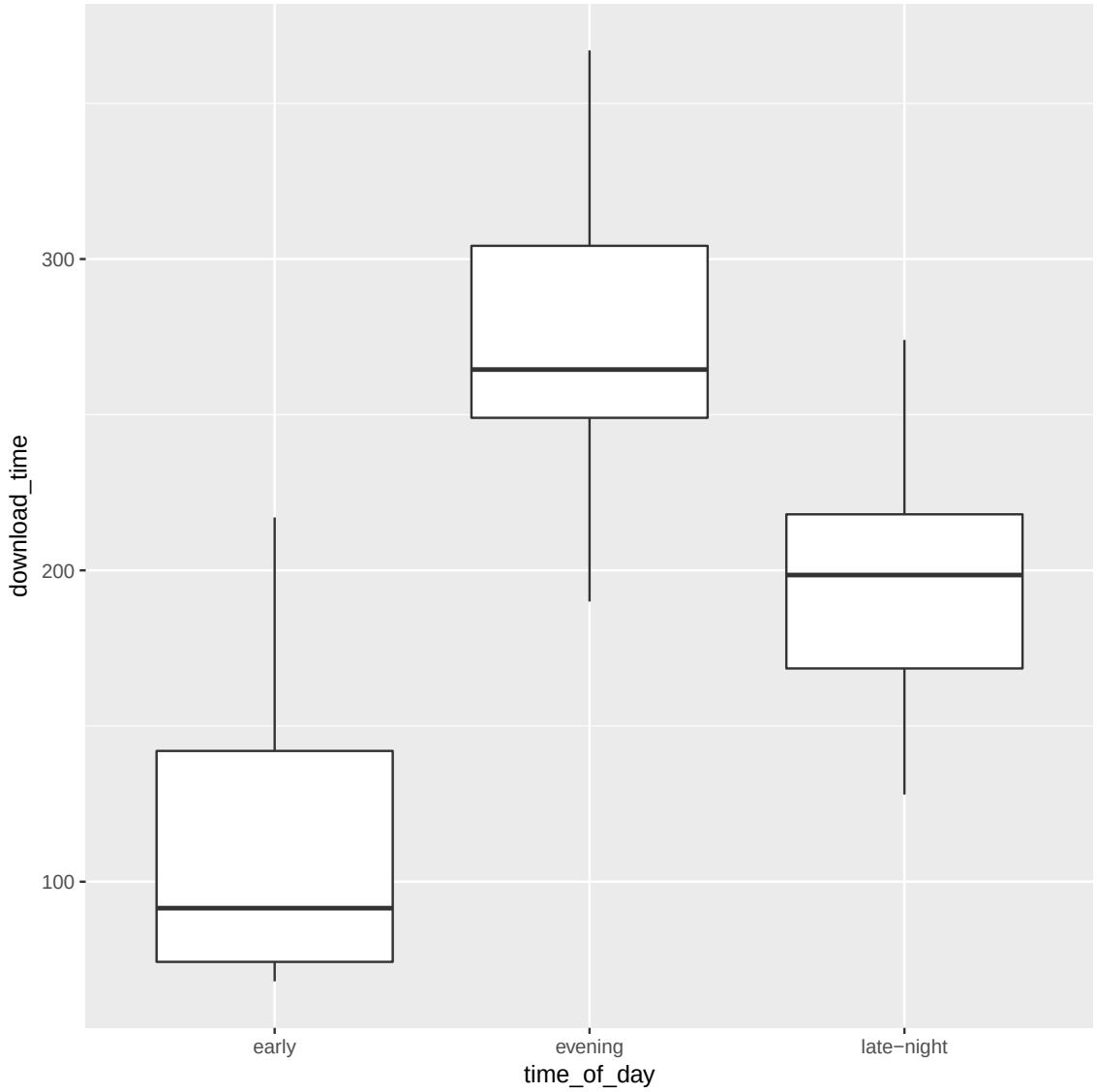
**Solution:**

```
ggplot(downloads,aes(x=time_of_day,y=download_time))+
    geom_boxplot()
```




Or, if you prefer, with the same result:

```
downloads %>% ggplot(aes(x=time_of_day,y=download_time))+
    geom_boxplot()
```




This is meant to be *dead easy*. Don't make it more difficult than it needs to be, for example:

- no quotes on the column names ("backticks" are OK, but unnecessary).
- no need for `factor` on `time_of_day`: this is already text, so will be treated as categorical without changes. (The time to worry about this is when you have a variable taking *number* values that you want to treat as categorical.)

Other observations:

- *Two* close-brackets on the end of `ggplot`, one to close the `ggplot` and one to close the `aes`. I let you get away with this if the code was otherwise correct.
- Use the variable names *as given in the data file*, Figure 1. This is what will get read in. If you want to use your own variable names, you'll need to put them in your `read_tsv`. (One student did this successfully and thus got full credit, but why confuse me when you don't have to?)

- `geom_boxplot` *must* have an open and close bracket afterwards. If you leave off the brackets, you get a cryptic error message:

```
ggplot(downloads,aes(x=time_of_day,y=download_time))+geom_boxplot
## Error:  Don't know how to add geom_boxplot to a plot
```

  This, computationally, is because you are adding *the function* `geom_boxplot` into `ggplot`, instead of a *call* to the function (with no non-default options).

If you had some code that looked correct for the `ggplot` part, or some code that would produce a boxplot, I gave you a point. Thus, 1 point covers a wide range of answers here, ranging from one smallish but important error down to one correct thing plus nonsense.

(c) (2 marks) Explain briefly why a boxplot is a sensible plot to draw for these data.

**Solution:** We have one quantitative variable, download time, and one categorical variable, time of day.

If you like, you can comment that side-by-side boxplots allow us to compare typical download times for different times of day. That will also work as an answer.

You need to tell me *which* variable is categorical and which quantitative. If you don't, you demonstrate only that you know the theoretical situation in which a boxplot applies, but not why the particular boxplot drawn in the previous part was the right thing. (I was willing to tolerate the names `x` and `y` for this, since I took these from your boxplot code.)

If you can tell me that boxplots are for comparing something reasonably sensible, that got you a mark.

It also pays you to understand the distinction between a categorical variable, such as `time_of_day`, which tells you which time of day each download was done, and its *levels*, the three different possible values (early, evening, late-night). The number of side-by-side boxplots is the number of *levels*, here 3, not the number of categorical variables, here 1.

(d) (2 marks) The boxplot drawn by your code is shown in Figure 2. Does there seem to be a good time of day to download the file? Explain briefly.

**Solution:** We are looking for a time of day when download times are typically *small*. That is clearly early morning. Download times at late night are typically longer, and in the evening are longer still.

Having said that, download times in early morning are rather variable, and sometimes an early-morning download takes longer than an evening download. But the overall picture is that the early morning will usually give the fastest download.

Don't expect any points if you picked out the *highest* time. Why would you be interested in the time of day where downloads take the *longest* time?

Strictly speaking, a boxplot doesn't show a mean, just a median (unless it's a SAS boxplot with a diamond on it). But I was willing to let that go if you could convince me you understood that the "typical" download time was smallest in the early morning, or that most of the fastest download times happened then.

2. The high jump has been an Olympic event since 1896 for men and 1928 for women. The winning height (in metres) in the high jump at each Olympic Games for men and women was recorded. The data set is shown in Figure 3. We are interested in making a graph that shows how the winning height changes over time. The data has been read into a data frame `highjump` and a SAS data set called `highjump` that is the most recently-created one.

   (a) (2 marks) What would be a suitable graph to display all the variables *on one graph*?

**Solution:** The variables are two quantitative (`Year` and Height) and one categorical (`Gender`). So the appropriate thing is a scatterplot with the points for each level of the categorical variable displayed differently, such as by colour.

I didn't ask for an explanation, so if you can leap to a good answer, that's fine. But bear in mind that if you give an explanation, it might be worth a point even if you are wrong. An answer of "a scatter plot" is wrong because it doesn't use all three variables, but an answer of "some kind of scatter plot, but I don't know how to put `Gender` on it" is worth one point, because you recognize what needs to be done, even if you don't know how to do it.

Because of the way I asked the question, "a scatterplot with the points distinguished by groups" was enough, but you would do well to get in the habit of identifying the two quantitative and one categorical variables that make this the right kind of graph. If I had instead asked "identify the type of each variable in the data set and thus say what would be a suitable graph to show them all", then I would have expected more detail from you.

This part is meant to guide you in thinking about what kind of graph you will be giving code for below. If you find it easier, think about the code first and come back and write this part later. But you need to say something to describe the graph you would draw.

I had some sympathy with treating year as categorical, and thus having grouped boxplots (with height as the only quantitative variable). But I don't like this as much because there is only *one* height for each year-gender combination, so you get one-observation boxplots (see below) which always look kind of dopey to me. (I think this is a sub-optimal *choice* of graph, so I gave it 1 out of 2 here, but you get full marks below for successfully *drawing* it.)

Any kind of bar chart would be no good because these are used when *all* your variables are categorical.

   (b) (3 marks) Give R code to draw your graph of part (a).

**Solution:** Simpler than you would think. Excuse me while I read in the data first:

```
highjump=read_csv("high_jump.csv")
```

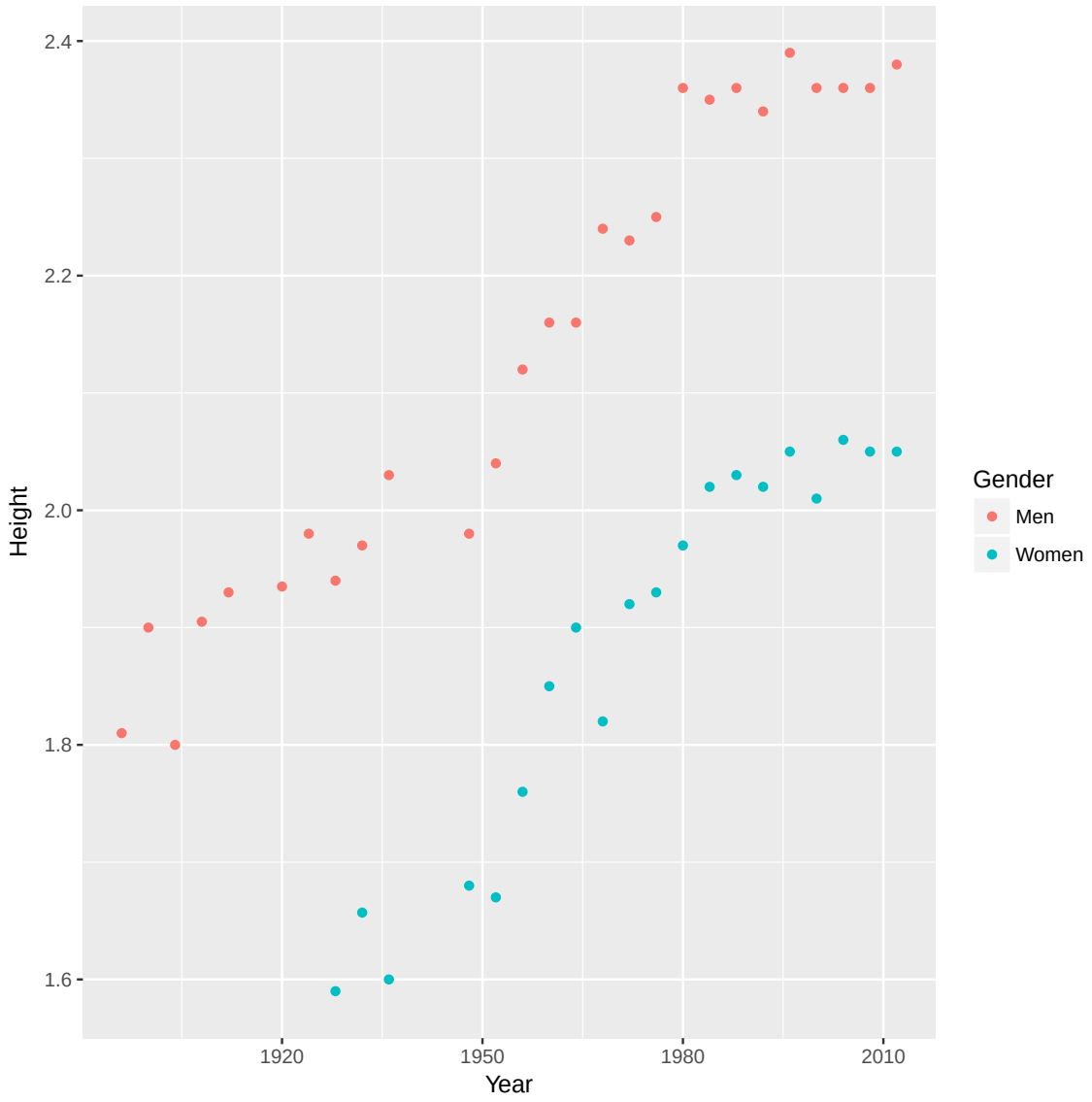
```
## Parsed with column specification:
## cols(
##   Year = col_integer(),
##   Gender = col_character(),
##   Height = col_double()
## )
```

```
highjump
```

```
## # A tibble: 47 x 3
##     Year Gender Height
##    <int>  <chr>  <dbl>
##  1  1896    Men  1.810
##  2  1900    Men  1.900
##  3  1904    Men  1.800
##  4  1908    Men  1.905
##  5  1912    Men  1.930
##  6  1920    Men  1.935
##  7  1924    Men  1.980
##  8  1928    Men  1.940
##  9  1932    Men  1.970
## 10  1936    Men  2.030
## # ... with 37 more rows
```

and then

```
ggplot(highjump,aes(x=Year,y=Height,colour=Gender))+geom_point()
```

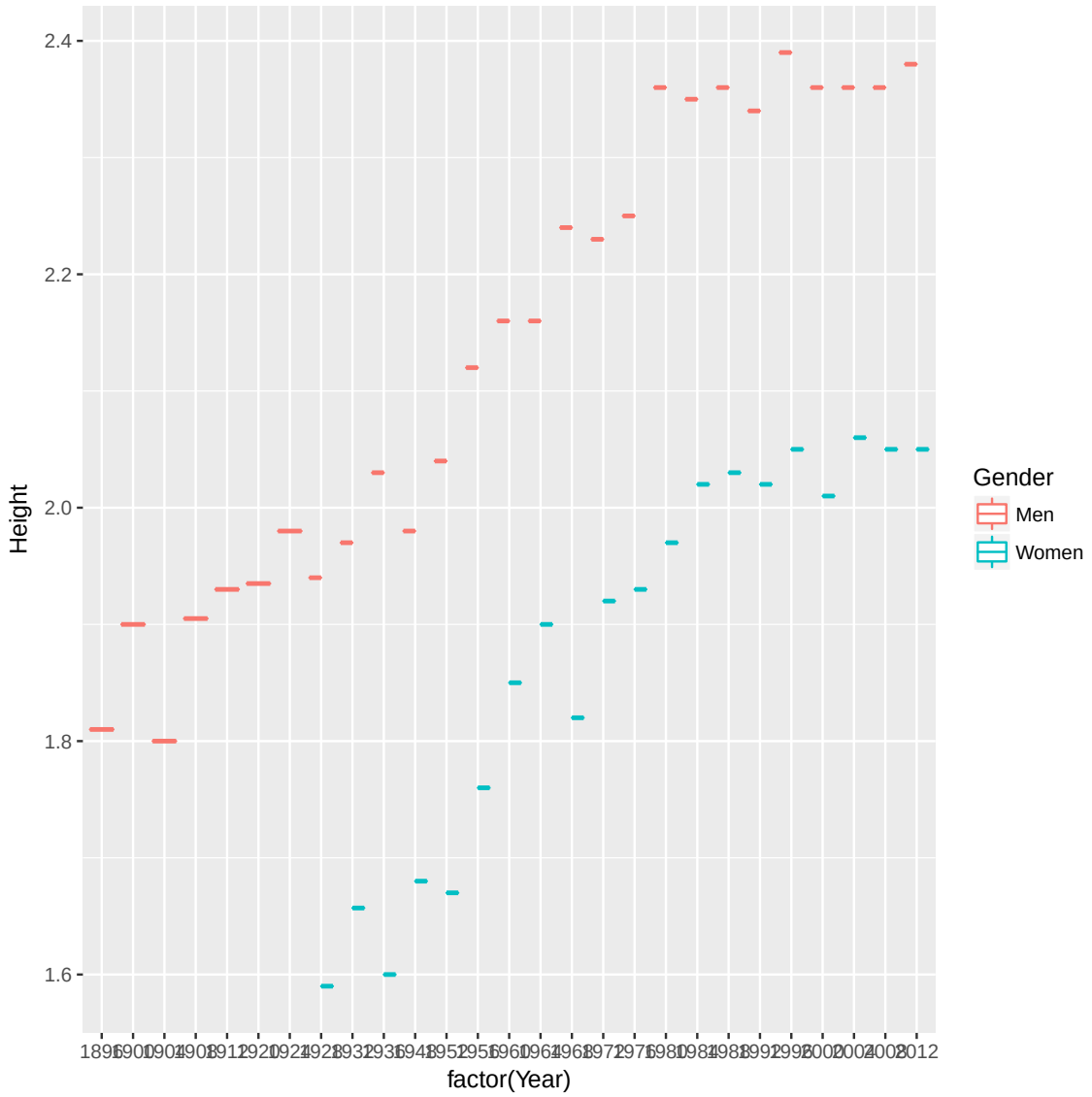



The variable names *must* have Capital Letters, because that's how they were in the data frame and R is case-sensitive. I was willing to forgive one variable without Uppercase, but if you got two or more of them wrong, that cost you a mark. Also, the year should be on the $x$-axis, because that is explanatory and the height is the response, or (if it makes more sense to you) time "always" goes on the $x$-axis.

A lot of people got the graph code right without properly describing the graph they were going to draw. A clue is to look at your code for (b) and go back to (a), checking that you have said everything you need to say, for example, explaining why you have a `colour=Gender` in there (that's the bit that colours the points differently for males and females). If you don't say that in (a), it makes me wonder whether you really understand what the code is doing.

I mentioned above that I didn't like grouped boxplots so much here:

```
ggplot(highjump,aes(x=factor(Year),y=Height,colour=Gender))+
geom_boxplot()
```




This is, I guess, not so bad, but in effect it's a scatterplot with a strange $x$-scale. (The years were numbers that we wanted to treat categorically, so we had to ask for `factor()` of them. Failing to do this if you drew boxplots cost you a point.) The place where this kind of plot would be useful is if you had multiple measurements of height for each gender-year combo, for example the heights jumped by *all* the finalists in the appropriate high jump competition each year, instead of just the winner.

If you thought a simpler graph was all that was required, you'll get some credit here for drawing it (successfully), but not full credit because you have made it simpler than it should be. For example, whatever graph you come up with has to include all three variables somehow, and if it doesn't, you can expect to lose some more points.

I saw a few attempted graphs with `aes(x=men, y=women)` in them. This shows a fundamental lack of understanding of the difference between a categorical variable and its levels. Sort this out now, or else when we get to tidying data, you're going to get very confused.

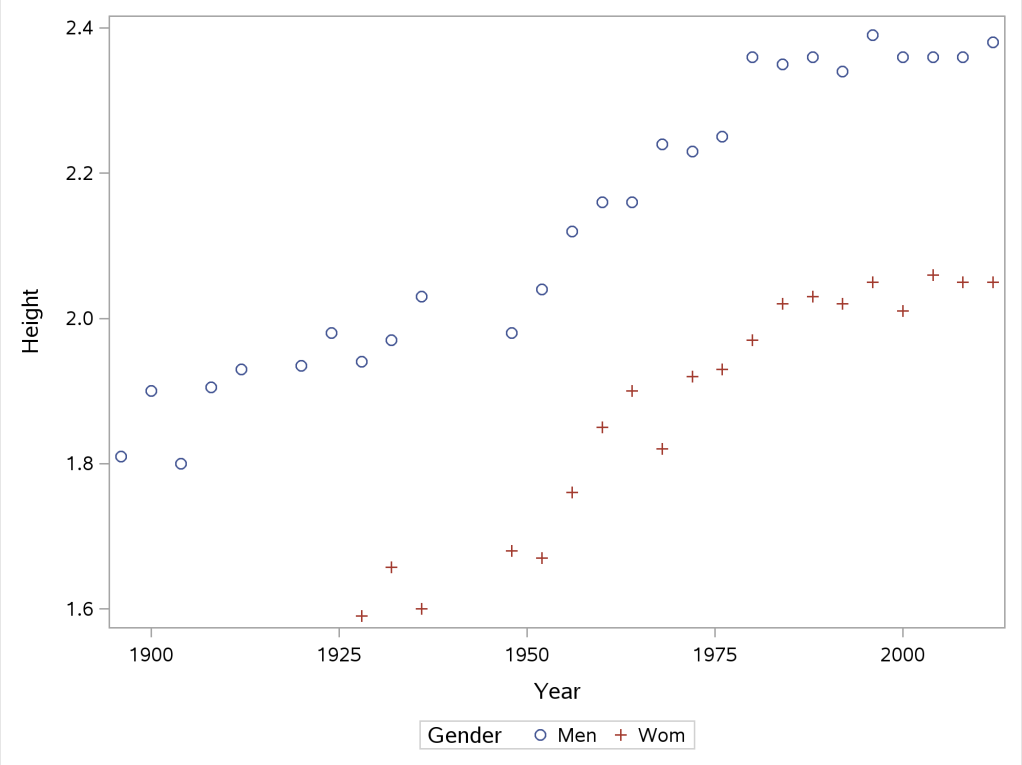
(c) (3 marks) Give SAS code to draw your graph of part (a).

**Solution:** Again, not that much to it. I have to read the data first, but you don't have to:

```
proc import
  datafile='/home/ken/high_jump.csv'
  out=highjump
  dbms=csv
  replace;
  getnames=yes;
```

Then the bit that you need. The data set was the most recent one, so `proc sgplot` without a `data=` is fine (though it doesn't hurt to add `data=highjump`):

```
proc sgplot;
  scatter x=year y=height / group=gender;
```

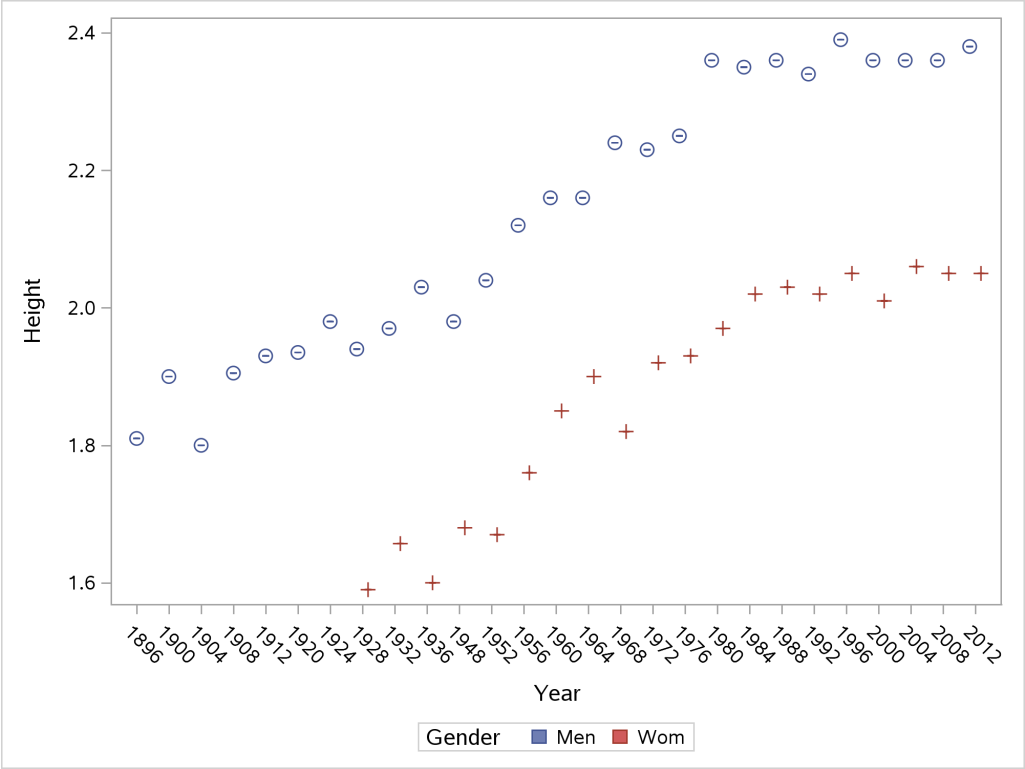


SAS is *not* case-sensitive, so you can use lower-case variable names here if you like (as I did), though of course there's no problem in calling the variables `Year` and `Height` and `Gender` again.

If you get the $x$ and $y$ the wrong way around, I won't penalize you again if you did that in the previous part. If you are inconsistent, expect to lose a mark: that is, getting the $x$ and $y$ axes the wrong way around once or twice will cost you one mark. (Having said that, I don't think anybody did this.)

If you went the boxplot way, I think the best SAS approach is this:

```
proc sgplot;
  vbox height / group=gender category=year;
```

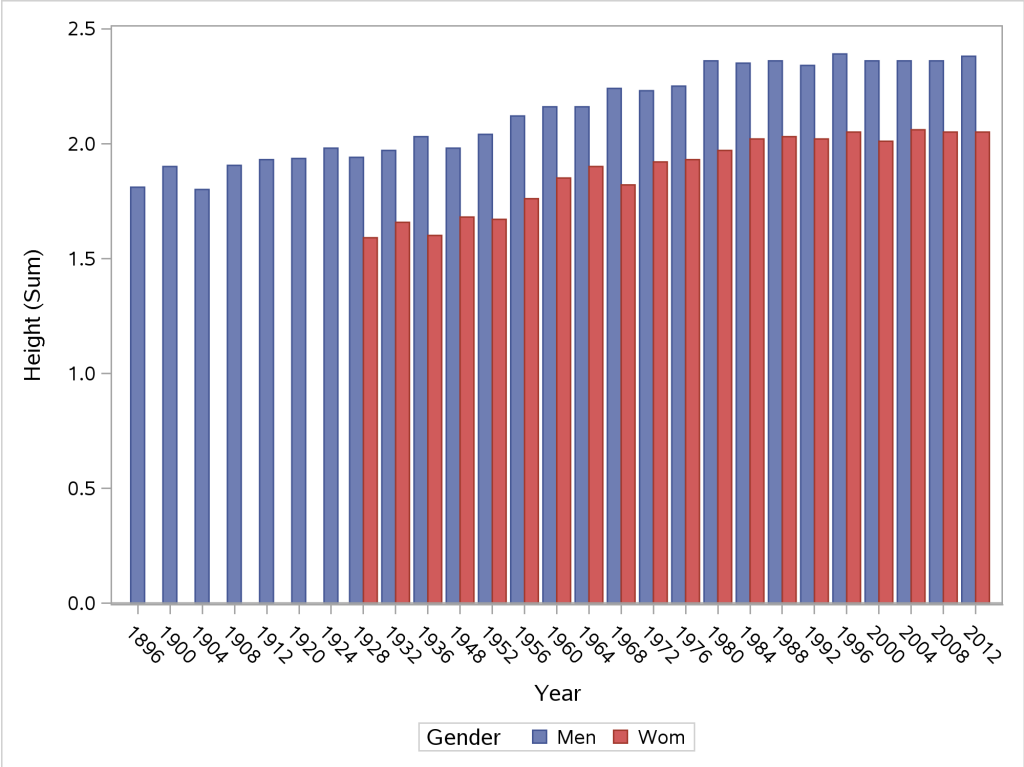


I'm not going to quibble if you switch the `group` and `category`, though the plot comes out less clear, to my mind.

There is no SAS equivalent of `factor()` required here: the plot requires `year` to be categorical, so it is automatically made such.

This also works, I think:

```
proc sgplot;
  vbar year / response=height group=gender groupdisplay=cluster;
```




I'd give full marks for this as shown.

This makes a bar chart for each year and gender, but instead of the usual frequency being the height of the bars, the `response=` uses the variable named as the heights of the bars. This works here because there is only one height for each year and gender; otherwise they get added up, which is not what you would want here. The final `groupdisplay=cluster` is needed to display the bars side by side for each year. Otherwise they come out stacked, which is very hard to understand.

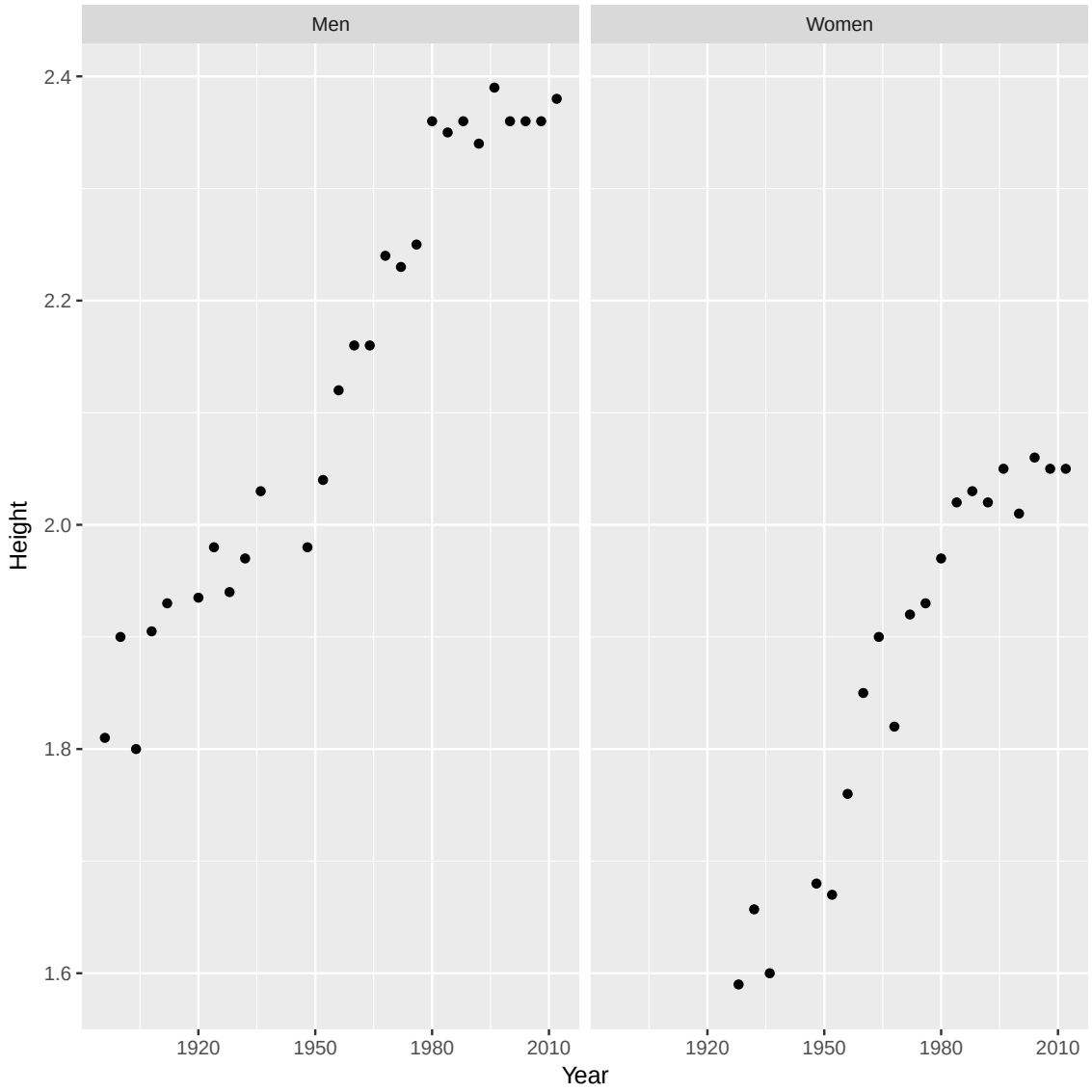
As I see it, if you have to cajole a bar chart to display what you want, that's a signal that you probably want some other kind of graph, with the implication that there is a much easier way to do it.

(d) (3 marks) Give R code for one plot *command* that produces two appropriate subplots, one for men and one for women. (By "subplots" I mean graphs like ones you made above, but one containing only men and one containing only women, contained within one larger plot.)

**Solution:**

The idea of one plot command making two subplots is meant to suggest facets. The idea is to make a scatterplot of all the heights and years together, and then add a `facet_wrap` or equivalent to do separate subplots for each gender:

```
ggplot(highjump,aes(x=Year,y=Height))+geom_point()+facet_wrap(~Gender)
```

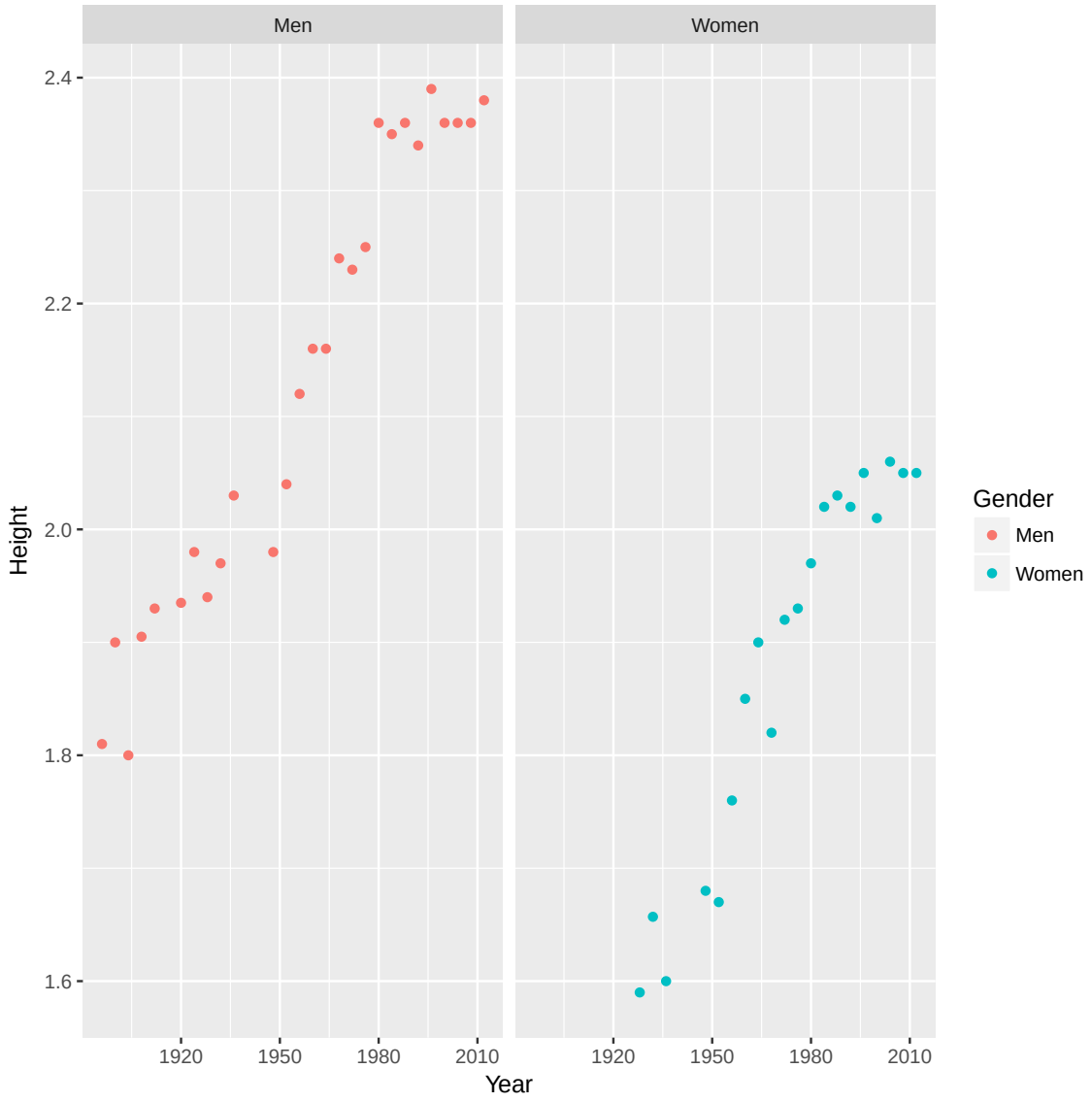



I have no objection if you add `scales="free"`, though it's not necessary. I actually think I want the height scales the same on both graphs, so I can compare the heights for men and women. But that's your call. Either way is good here.

My guideline was 2 points for mentioning `facet_wrap`, though you could get only one this way by making two other mistakes. If you didn't mention `facet_wrap`, your maximum was one point, and to get that you had to demonstrate some improvement on your part (b). For example, you might have failed to mention `colour=Gender` in (b), but if you then mentioned it in (d), I was prepared to give you a point

there. I want to see you *remove* the `colour=Gender` from your answer to (b); since we are distinguishing the genders by putting them in different facets, there's no need to distinguish them by colours as well:

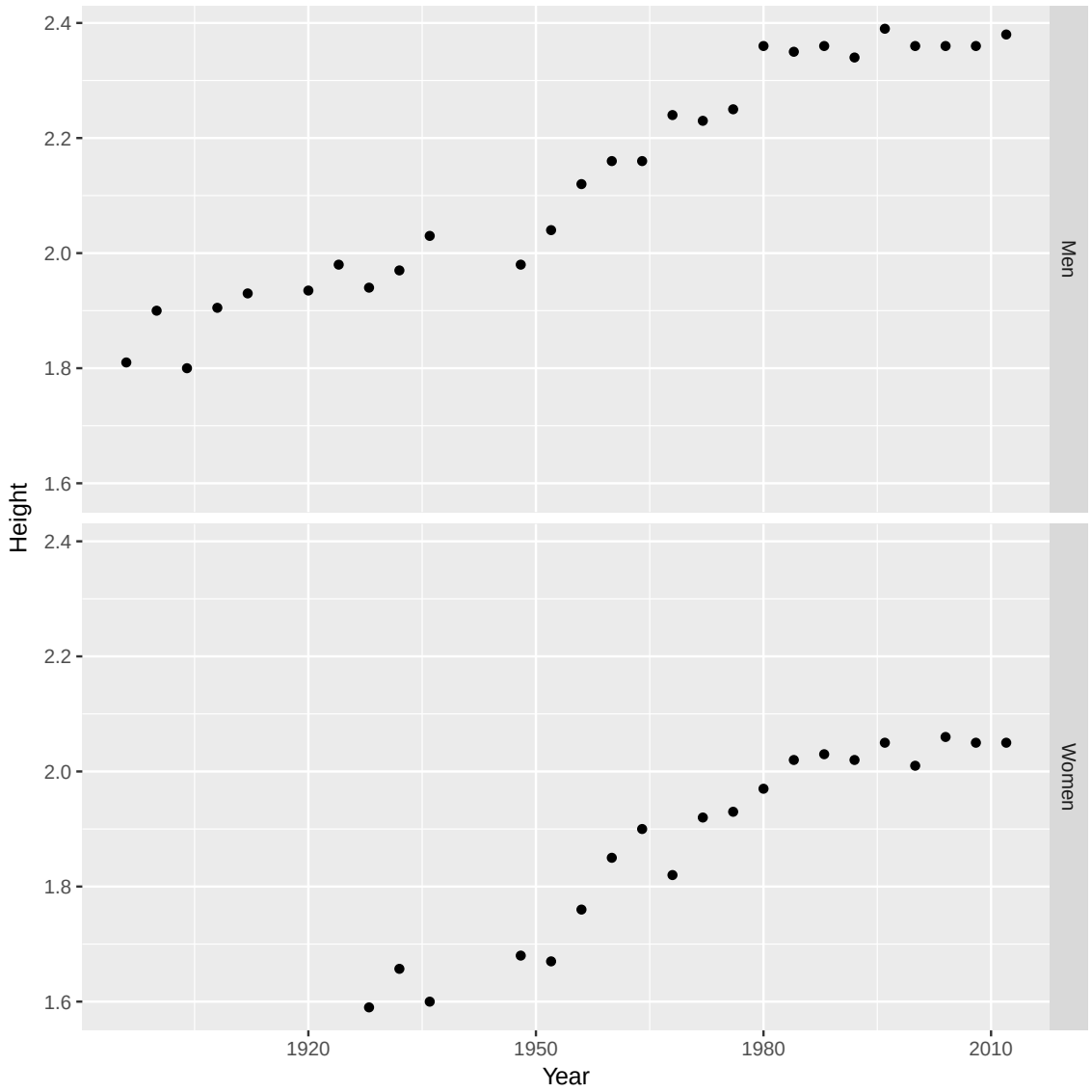
```
ggplot(highjump,aes(x=Year,y=Height, colour=Gender))+
  geom_point()+
  facet_wrap(~Gender)
```




It is rather pointless to have all the points on one subplot red and all those on the other one blue.

You could also use `facet_grid`, but you need to specify whether the two subplots are above and below or left and right. This is above and below, with the facets in the $y$-direction and "nothing" in the $x$-direction:

```
ggplot(highjump,aes(x=Year,y=Height))+geom_point()+facet_grid(Gender~.)
```




I don't like this as much, because the aim of the plot is to show how men and women compare, and it's easier to do that from the side-by-side plots. But it answers the question, so from that point of view I'm fine with it.

(e) (3 marks) Give SAS code, that calls *one* `proc`, that produces two appropriate subplots, one for men and one for women.

**Solution:** This is an invitation to use `proc sgpanel`. This begins with a `panelby` line to make the subplots by gender, and then continues with whatever you would use to make an `sgplot` for all the highjump winning heights together:

```
proc sgpanel;
```

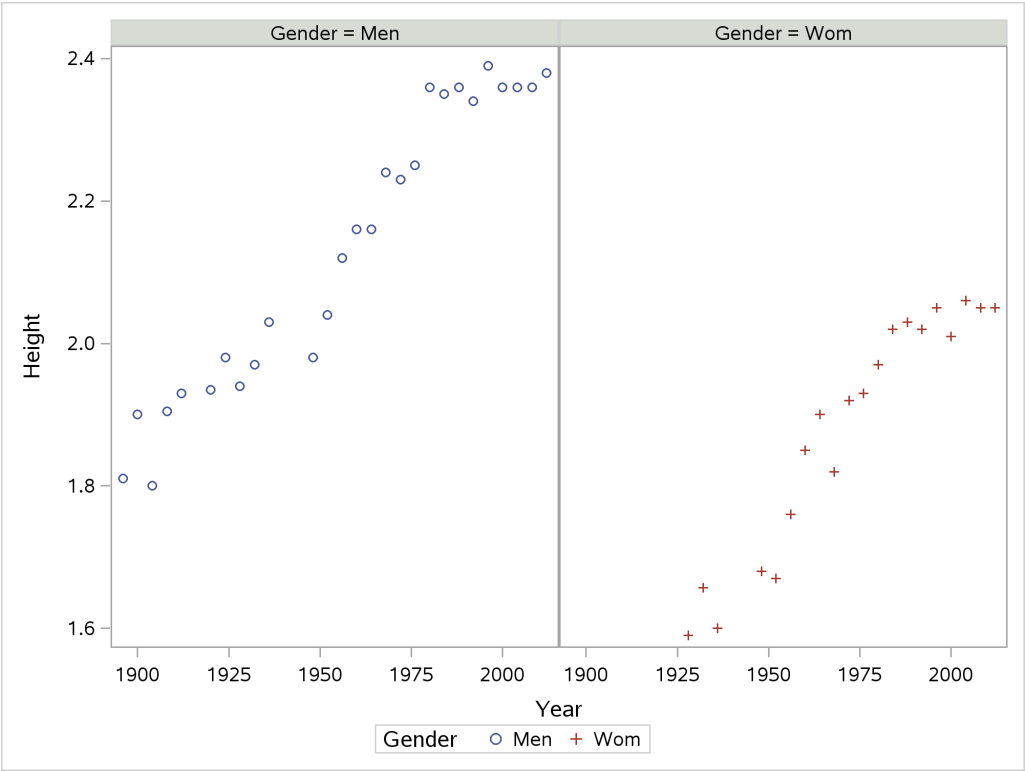
```
panelby gender;
scatter x=year y=height;
```


Gender = Men
Gender = Wom
Height
2.4
2.2
2.0
1.8
1.6
1900
1925
1950
1975
2000
Year

This is actually simpler than the one I had in lecture because the subplots are "straight" scatterplots, so there is no longer a `group` within the scatterplots. (The example we had in lectures had *four* variables, so one of them, gender, was still in the scatterplots and the extra categorical variable, sport, was defining the panels. Here, we only have three variables, so the subplot scatterplots only have two quantitative variables each.)

If you try to mimic the lecture example too closely, you'll get something like this (I seem to need the `data=` but you don't):

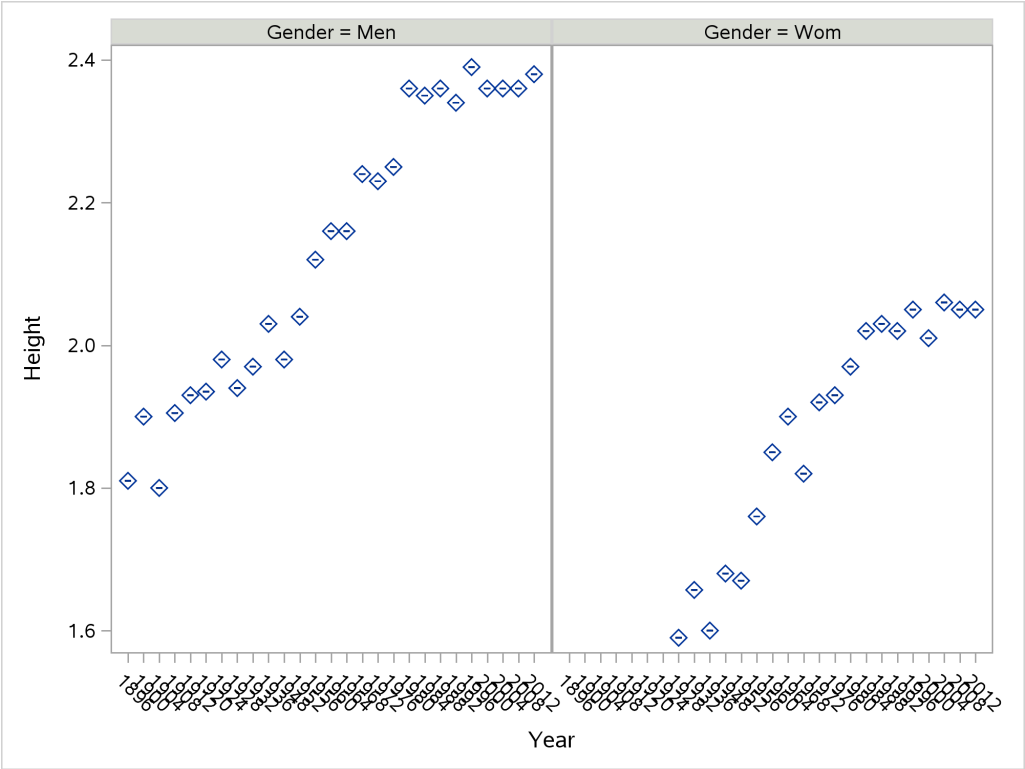
```
proc sgpanel data=highjump;
  panelby gender;
  scatter x=year y=height / group=gender;
```


Gender = Men
Gender = Wom
Height
2.4
2.2
2.0
1.8
1.6
1900
1925
1950
1975
2000
Year
Gender
Men
Wom

This shows a lack of understanding, because the *reason* for having the two separate subplots was to distinguish by gender, so there is no need to do it again.

If you thought boxplots were the answer (because year was categorical), you can get full marks by doing it like this (again the `data=` is not needed by you):

```
proc sgpanel data=highjump;
  panelby gender;
  vbox height / category=year;
```




As in the previous part, if you managed to recognize that `proc sgpanel` with `panelby gender` was the answer, I tried hard to give you at least two points (the exception being if you made two errors, as above). If you didn't get to `proc sgpanel`, you were fighting for one point. My guiding principle was whether you had improved on your answer to (c); for example, if you had missed out on plotting gender in (c) but you did with `category=gender` in (e), I thought you deserved one point in (e). Two very fast ways to get zero points in (e):

1. copy your code from (c) with *no changes*.
2. two `proc sgplots` one after the other. (That's why I said "one `proc`".)

3. In psychology, the "Stroop test" is a test in which colour names are printed in a different colour. For example, the word "red" might be printed in blue ink. The idea of the test is that you name the colour *of the ink* as fast as you can (for a whole collection of colour names printed in different colour ink). There is a mental conflict involved: when you see "red" printed in blue ink, you have to stop yourself saying "red" and say "blue" instead.

   A researcher thinks that hypnotizing people helps them to do better on the Stroop test, and so conducts an experiment on twelve subjects. Six of these subjects, randomly chosen, do the Stroop test as normal (without being hypnotized), while the other six are hypnotized first. The score on the test is a combination of the time taken and the number of mistakes made; a lower score is better. The data are shown in Figure 4. I saved the data as `stroop.txt` under my username `ken` on SAS Studio.

   (a) (4 marks) Give SAS code to read in and display the data.

**Solution:** This is `proc import`. Looking at the file, the data values are separated by exactly one space, so `dbms=dlm` will do the job:

```
proc import
  datafile='/home/ken/stroop.txt'
  out=stroop
  dbms=dlm
  replace;
  getnames=yes;
  delimiter=' ';

proc print;
```

This produces output:

| Obs | subject | hypnotized | score |
|---|---|---|---|
| 1 | 1 | yes | 8.5 |
| 2 | 2 | yes | 9.6 |
| 3 | 3 | yes | 10 |
| 4 | 4 | yes | 9.2 |
| 5 | 5 | yes | 8.9 |
| 6 | 6 | yes | 10.8 |
| 7 | 7 | no | 12.6 |
| 8 | 8 | no | 13.8 |
| 9 | 9 | no | 11.6 |
| 10 | 10 | no | 12.2 |
| 11 | 11 | no | 12.1 |
| 12 | 12 | no | 13 |

which looks like the original data file.

You can use a `filename` if you like, but it's different from the ones we used in asssignments to read data from URLs (the data here is already in my/your account on SAS Studio). You still need `/home/ken` or equivalent on the file name:

```
filename myfile '/home/ken/stroop.txt';

proc import
  datafile=myfile
  out=stroop
  dbms=dlm
  replace;
  getnames=yes;
  delimiter=' ';

proc print;
```

| Obs | subject | hypnotized | score |
|---|---|---|---|
| 1 | 1 | yes | 8.5 |
| 2 | 2 | yes | 9.6 |
| 3 | 3 | yes | 10 |
| 4 | 4 | yes | 9.2 |
| 5 | 5 | yes | 8.9 |
| 6 | 6 | yes | 10.8 |
| 7 | 7 | no | 12.6 |
| 8 | 8 | no | 13.8 |
| 9 | 9 | no | 11.6 |
| 10 | 10 | no | 12.2 |
| 11 | 11 | no | 12.1 |
| 12 | 12 | no | 13 |

If you *must*, use a data step, but that rather betrays that you have not been coming to class:

```
data stroop;
  infile '/home/ken/stroop.txt' firstobs=2;
  input subject hypnotized $ score;

proc print;
```

| Obs | subject | hypnotized | score |
|---|---|---|---|
| 1 | 1 | yes | 8.5 |
| 2 | 2 | yes | 9.6 |
| 3 | 3 | yes | 10.0 |
| 4 | 4 | yes | 9.2 |
| 5 | 5 | yes | 8.9 |
| 6 | 6 | yes | 10.8 |
| 7 | 7 | no | 12.6 |
| 8 | 8 | no | 13.8 |
| 9 | 9 | no | 11.6 |
| 10 | 10 | no | 12.2 |
| 11 | 11 | no | 12.1 |
| 12 | 12 | no | 13.0 |

Expect to lose one mark per error down to a minimum of 1 if you got *something* right. (Hint: you will lose marks more quickly with a `data` step, since it is easier to get things wrong). Common places to lose marks:

- forgetting the `/home/ken/` on the filename. Using `/folders/myfolders/` is good, *if* you get it right!
- forgetting the `delimiter` line. SAS has to know what the values are separated by!
- forgetting the `proc print` to display the values. To those of you who forgot this: *read the question all the way through.* Sloppiness and statistics do not mix well.

(b) (2 marks) What kind of plot would be most suitable to display these data? We do not want to display the subject numbers on the graph.

**Solution:** Apart from the subject numbers, we again have one categorical variable (whether or not the subject was hypnotized) and one quantitative one (the score on the Stroop test). So this is again a boxplot.

I didn't ask for an explanation (maybe I should have done), so the single word "boxplot" is enough for the two points. I was happy to see, though, that a lot of people added an explanation, including some that were very clear. (As before, you can note the one quantitative and one categorical variable, or you can say something along the lines of comparing the scores for the people who were or were not hypnotized.)

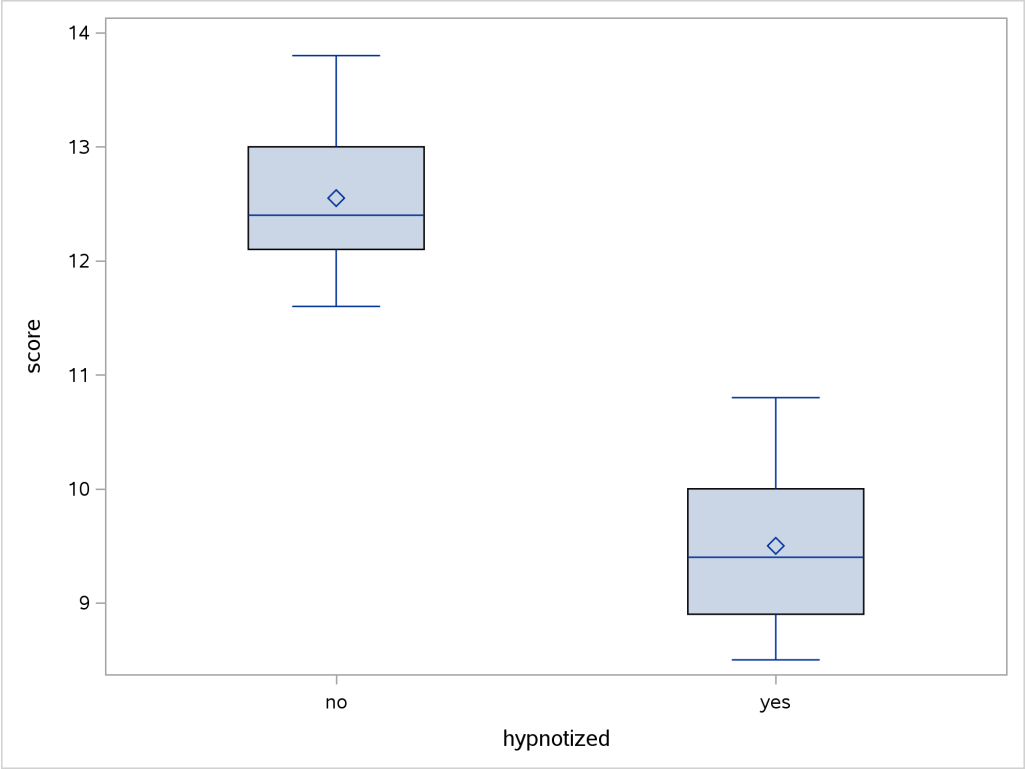
Another possible plot is panelled histograms (of scores for the hypnotized and non-hypnotized people side by side). I'm happy with this, but the word "histogram" by itself is only one mark because you need the extra detail of "panelled".

I can't see a way that a scatterplot would work here, because there is only one quantitative variable, even if you count the subject numbers (which are labels rather than quantitative).

(c) (2 marks) Give SAS code to draw the plot that you proposed in the previous part.

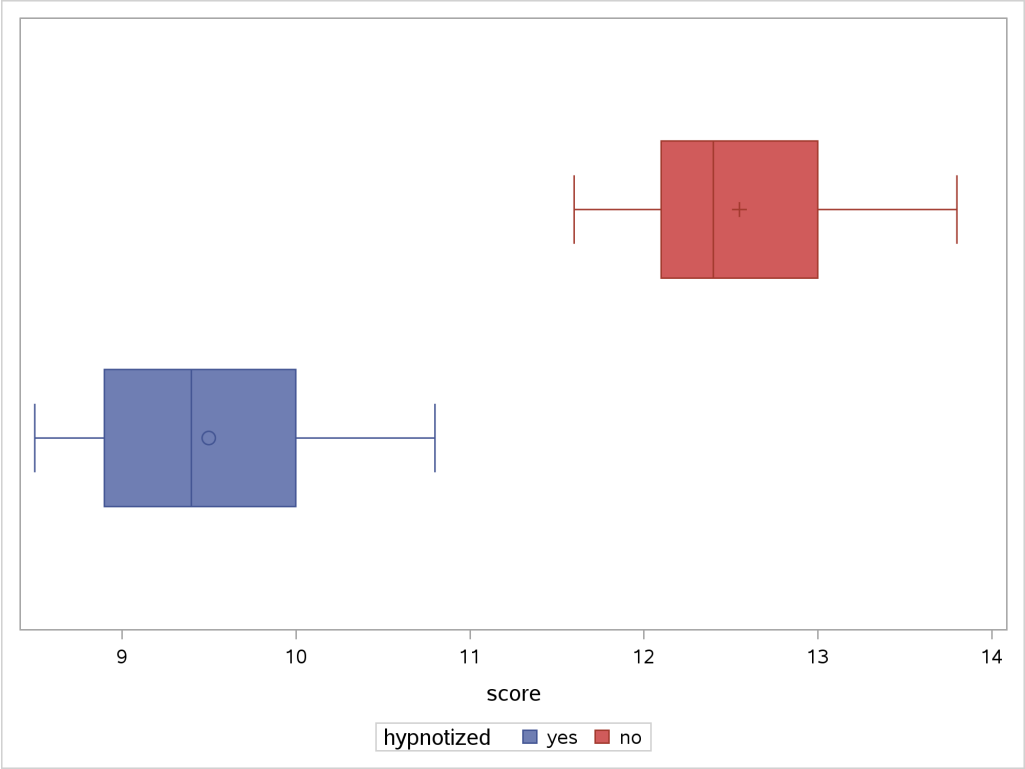
**Solution:** Assuming that you too thought that a boxplot was the thing:

```
proc sgplot;
  vbox score / category=hypnotized;
```



A horizontal box was equally good, or using `group` instead of `category`, eg:

```
proc sgplot;
  hbox score / group=hypnotized;
```

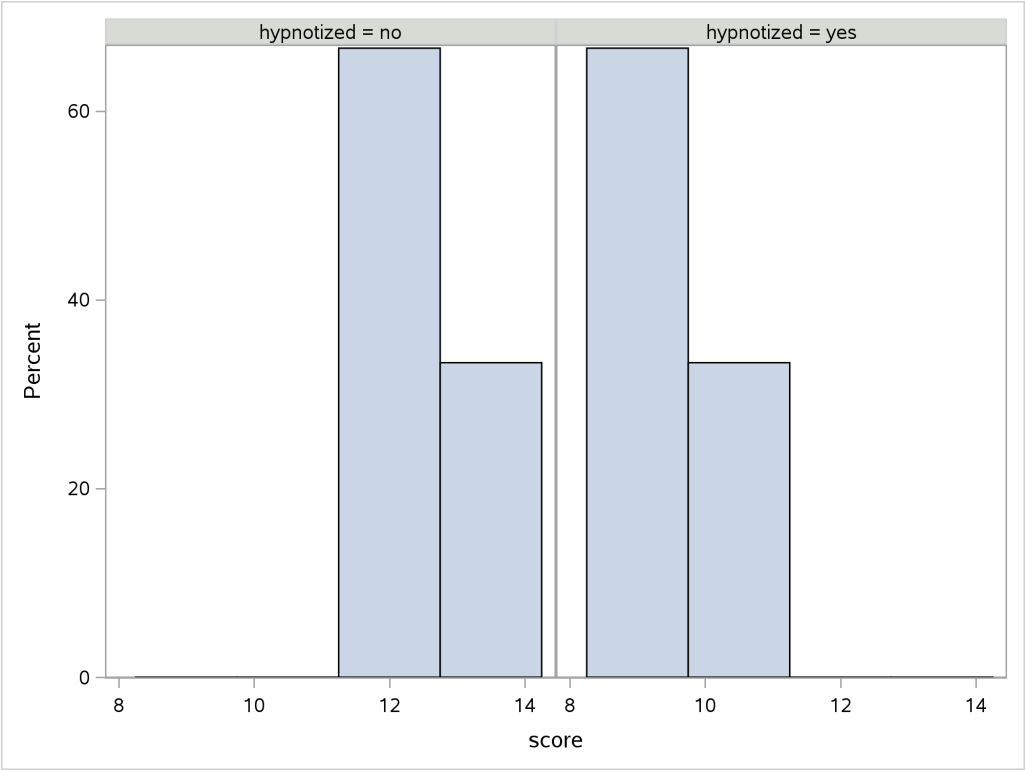


Using `group` instead of `category` colours the boxplots.

Or, panelled histograms:

```
proc sgpanel;
  panelby hypnotized;
  histogram score;
```

That's a lot of work for two points, but anyway:




If I disagreed with your choice of plot, I tried hard to be sympathetic to your attempt to draw what you described.

Apart from that, though, I tried to reserve 1 out of 2 here for people who got close to drawing a sensible plot, eg. with one small error. Since I didn't ask for an explanation of your choice of boxplot (if that's what you chose), you might have chosen a boxplot for the wrong reason, in which case you got a lucky two points in (b) but will then have been penalized in (c) for doing something less than sensible there. (I can't guarantee that I looked at explanations in (b).) This part's fast route to zero was to use `subject` somehow in your plot (even though I said I didn't want `subject` in the plot), or to use a variable like `colour` that wasn't in the data set. (There would have been another spreadsheet or similar that had the colour names and the ink colour, along with a list of responses for each subject and time taken, and probably a calculation of each subject's score, but we never saw that. All we saw were the calculated scores.)

Next year, I'm going to have to make a bigger deal about how to plot identifier variables. Typically what is done is to use them as labels for points on plots. We learn about labelling on plots later.

(d) (3 marks) Give SAS code to carry out a suitable $t$-test to determine whether the hypnotized subjects

do better at the Stroop test on average than the unhypnotized ones. If you think your test should be one-sided, justify your choice of side.

**Solution:**

This requires a little thinking first before you write any code. A better score on the Stroop test is a lower one. The two values of `hypnotized`, from looking at the data in Figure 4, are `yes` and `no`, but what matters is the first one *alphabetically*, not the first one in the data. That means `no` is first. So, to get the alternative hypothesis right, we want the scores for `no` to be *bigger* than the scores for `yes`, on average:

```
proc ttest sides=U;
  var score;
  class hypnotized;
```

`side=U` is equally good, and one letter fewer to type (or write). I want to be sure that you picked the upper tail for the right reason. For example, thinking that "yes" comes first and a high score is better gets the right answer for the wrong reason, and so will lose marks. In fact, this thinking makes two errors, so you were in one sense lucky not to lose two marks for it. (In the end, I counted this as only one error.)

Doing it right produces output

```
hypnotized     N        Mean     Std Dev     Std Err     Minimum     Maximum

no             6     12.5500      0.7740      0.3160     11.6000     13.8000
yes            6      9.5000      0.8246      0.3367      8.5000     10.8000
Diff (1-2)            3.0500      0.7997      0.4617

  hypnotized    Method               Mean       95% CL Mean        Std Dev

  no                              12.5500     11.7378  13.3622      0.7740
  yes                              9.5000      8.6346  10.3654      0.8246
  Diff (1-2)    Pooled             3.0500      2.2132  Infty        0.7997
  Diff (1-2)    Satterthwaite      3.0500      2.2128  Infty

              hypnotized    Method             95% CL Std Dev

              no                               0.4831   1.8982
              yes                              0.5147   2.0225
              Diff (1-2)    Pooled             0.5588   1.4034
              Diff (1-2)    Satterthwaite

         Method           Variances        DF    t Value    Pr > t

         Pooled           Equal            10       6.61    <.0001
         Satterthwaite    Unequal      9.9601       6.61    <.0001

                          Equality of Variances

             Method      Num DF    Den DF    F Value    Pr > F

             Folded F         5         5       1.14    0.8927
```

This was painful to mark, but the scale wound up as:

- 3: properly justified one-sided test with correct tail and $t$-test correctly coded.
- 2: correctly coded two-sided test or correctly coded one-sided test without complete justification of side
- 1: what would have been a 2-mark answer apart from errors, or something correct beyond `proc ttest`.
- 0: anything else.

If you managed to think this through clearly in exam conditions, you deserve your three points!

(e) (3 marks) The plot you drew earlier shows approximate normality and approximately equal spread for the test scores within each group, of people who were hypnotized and people who were not. The SAS output for your $t$-test is shown in Figure 5. What do you conclude from it, in the context of the data? Justify briefly any choices you make.

**Solution:** The first sentence justifies (i) using the $t$-test at all ("approximate normality", ie. normal enough) and (ii) using the pooled version of the $t$-test, since the spreads are approximately equal. So we should look at the "pooled" P-value, which is less than 0.0001. We have no doubt about rejecting the null hypothesis of equal means in favour of a one-sided alternative. That is to say, the subjects who underwent hypnosis have a smaller, or better, score than the subjects who didn't. In short, hypnosis helps.

I am also willing to entertain a choice of the Satterthwaite test, for an appropriate reason: something like "the Satterthwaite test works well even if the two groups have the same spread" would do it. But not this: "the two groups need to have exactly the same spread to use the pooled test, and they have only approximately the same spread. Therefore I use Satterthwaite." What this answer fails to understand is that even if the *population* spreads are exactly the same, the *sample* spreads probably will not be. Therefore the pooled test is a reasonable choice if the sample spreads, the only ones you can observe, are "close to equal". This is, of course, a judgement call, one that you have to make.

But you need to choose one test and say why. Just saying "the P-value is less than 0.0001" is not a complete answer, because you are not telling me which test you are doing. In this case it doesn't matter, but in other cases it might be important.

There's a clue in the output in Figure 5: if you look at the confidence intervals for the differences in means, you see that they both go up to plus-infinity. This means that the test that was done was upper-tailed. Making the further assumption that I didn't trick you by deliberately giving you the wrong test (would I do that?), you can infer that your code for the test ought to have had `sides=U` in it. But it doesn't say why. That, you have to work out.

So the mark scale for this one is:

- 1 for a properly reasoned choice of test (could be either Satterthwaite or pooled) or for noting that it doesn't matter which one we use since the P-values are the same
- 1 for rejecting the null in favour of the alternative that you had because of the small P-value
- 1 for saying something about the effect of hypnosis on Stroop test scores (you can say something like "improves" if you don't want to commit to "increases" or "decreases"; if your test was two-sided, you need a word like "affects" that does not imply direction). I was fairly relaxed about whether you ended up concluding it was greater or less, as long as you were at least somewhat consistent with what you'd done before.

Add up the points you earned. If you skip over the statement of rejecting the null, you'll still get the point for it if it is clear that you have understood it (eg. if you say something like "we have strong evidence that hypnosis improves Stroop test scores", which implies that a null hypothesis of no-improvement has been rejected).

The "folded F" test at the bottom of the output, the one with a P-value of 0.8927, is testing that the two groups have equal variance. I prefer not to use a test like this for deciding whether to use pooled or Satterthwaite (I'd rather just eyeball the spreads), but I have no objection if you base your decision on that, since then at least you're basing your decision on *something* relevant. My take is that tests of equality of variances tend to make it too hard to declare the variances different when the samples are small, and too easy when they are large. This is statistical significance vs. practical importance again; if you have large samples, the two variances could be significantly different even though they are close together, in a situation where using the pooled test would be fine. The flip side of this is the kind of situation we have here with two small samples; the SDs could be quite dissimilar without being significantly different. Here, though, the sample SDs are very close, and the folded F is a long way from significance, which is all consistent.

This is part of an actual Stroop test (as long as you are viewing this in colour):

| | | |
|---|---|---|
| BLUE | GREEN | YELLOW |
| PINK | RED | ORANGE |
| GREY | BLACK | PURPLE |
| TAN | WHITE | BROWN |

For the test, you have to look at the first row and say "red blue red" as fast as you can, and then go on to the next line ("blue green pink"). It's very difficult. (I've done it.) Apparently the theory about hypnotizing is that this can train you to "see" the colour names (that you have to ignore) as random letters, so that they are less of a distraction: they no longer look like colour names and thus no longer distract you.

I am now wondering whether the Stroop test is easier for people who don't have English as a first language: can such people "tune out" the colour names and just see the colours? (I guess this is easier if your first language uses a different *alphabet*.)

4. In lecture, we looked at the Australian athletes data set. This contains information on 202 athletes who play various different sports. Thirteen variables are measured for each athlete. Some of the data set is shown in Figure 6.

   (a) (2 marks) The variable `BMI` is the "body mass index". It is the ratio of height to weight-squared, and is often used as a measure of whether a person is over-weight or under-weight for their height. Give R code to obtain a 90% confidence interval for the mean BMI of all athletes (of which these are a sample). Your code can obtain other things as well as the confidence interval, but must obtain the appropriate confidence interval among its output. (The data frame is called `athletes`, and has already been read into R.)

**Solution:** Since we want a CI for the mean, we are looking at a one-sample $t$:

```
t.test(athletes$BMI, conf.level=0.90)
```

```
##
##  One Sample t-test
##
## data:  athletes$BMI
## t = 113.92, df = 201, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 90 percent confidence interval:
##  22.62291 23.28887
## sample estimates:
## mean of x
##  22.95589
```

or

```
with(athletes,t.test(BMI, conf.level=0.90))
```

```
##
##  One Sample t-test
##
## data:  BMI
## t = 113.92, df = 201, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 90 percent confidence interval:
##  22.62291 23.28887
## sample estimates:
## mean of x
##  22.95589
```

or even

```
athletes %>% pull(BMI) %>% t.test(conf.level=0.90)

##
##  One Sample t-test
##
## data:  .
## t = 113.92, df = 201, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 90 percent confidence interval:
##  22.62291 23.28887
## sample estimates:
## mean of x
##  22.95589
```

I haven't shown you the last one. `pull` means "pull out just the column named". Then the BMI values get fed into `t.test` as the first thing (which thus is not named explicitly).

One way or another you need to say "the BMI that lives in data frame `athletes`". This is a one-sample $t$-test, so it *does not* work with `data=`. The wording in the question was intended to make clear that you can get other things as well as a confidence interval (a $t$-test, specifically). You don't need a null mean (if you only want a CI), but it doesn't hurt to supply one (I ignored it if you did). If you tried `data=` and that was your only mistake, you got 1.

Not much else to say. I did not ask for a two-sample CI by gender, or for a CI for the median, or anything like that.

(b) (2 marks) The 90% confidence interval found by your code in the previous part goes from 22.6 to 23.3. One of the three interpretations below is the best one. Which one? Explain briefly: (i) the procedure by which this confidence interval was produced would give you an interval containing the population mean BMI for 90% of all possible samples; (ii) 90% of all athletes have BMI values between 22.6 and 23.3; (iii) the interval from 22.6 to 23.3 has probability 0.90 of containing the population mean BMI.

**Solution:** It's (i), because the confidence is in the *procedure* by which the interval is produced, not in any one interval. If you prefer, you can rule out the other two: (ii) is no good because we are talking about the *mean* BMI of all athletes, not the distribution of all values (which will go up and down a lot further), and (iii) is no good because the interval we got either *does* or *does not* contain the population mean: we just don't know which.

Said differently, the randomness comes from the idea of repeated sampling: "if we were to take another sample, what might happen?", so we cannot make any probabilistic statement based on only one interval.

This comes from your prerequisite course, STAB57 or STAB27 (it's actually in STAB22), and so *you ought to know it.*

This was a hard one to mark because of the variety of answers. My basic grading principle was to give 2 for a satisfactorily-reasoned answer, 1 for one sensible point otherwise, and 0 if I didn't think you got at any of the issues. I realized that an answer like "(i) because this is how a confidence interval is defined" doesn't display much understanding, but it *does* answer the question, so I had to give it two points (especially if combined with an assertion that "(ii) and (iii) are not"). I also saw some answers ruling out (iii) because "a confidence interval is not a probability", which is true but less insightful than saying that the population median either is or is not in $(22.6, 23.3)$.

You might find this definition of confidence interval unsatisfactory, but it's the way confidence intervals

are. If you've studied Bayesian inference, you'll know that it treats the population mean $\mu$ as a *random variable* (rather as something fixed but unknown). What you do there is to propose a "prior distribution" (your knowledge about $\mu$ before you collect any data) and calculate the "likelihood" of your observed data, and these get combined into a "posterior distribution" that encompasses *all* your knowledge about $\mu$. Thus, if you have a posterior distribution, you really *can* say that there is a 90% probability that $\mu$ lies between this value and that one, which is, you might argue, what you really wanted in the first place. Such an interval is called a "90% credible interval".

R has a package called `rstan` for doing Bayesian inference: what it does is to *sample* randomly from the posterior distribution, so that by taking many random samples, you can approximate the posterior distribution as closely as you like. Going this way saves you the need to do some algebra and recognize the distributional form of the posterior distribution (which, depending on your problem, is somewhere between tedious and impossible). If you have sampled your posterior distribution, you get a 90% credible interval by taking the middle 90% of the sampled values (that is, finding their 5th and 95th percentiles).

(c) (3 marks) A histogram is shown in Figure 7. Based on this histogram, do you have any doubts about the confidence interval calculated above? Explain briefly. What, if anything, would you do instead?

**Solution:** This histogram is skewed to the right, as I see it: there are a few very big BMI values. This would be enough to cast doubt on using the $t$ interval, which requires approximately normal data. You can say that we could use the CI but we should be cautious about it, or that we should compute a CI for the median instead (from the sign test).

You can also make the case that we have a big sample ($n = 202$) and something like "mild skewness" (as opposed to "bad skewness" or "serious skewness"), and therefore that the Central Limit Theorem will come to our rescue. If you go this way, you should say that the CI for the mean based on the $t$-distribution is reasonable, and that we don't need to do anything else.

Make a call. If you wanted a CI for the median, you could do it like this:

```
library(smmr)
ci_median(athletes, BMI, conf.level=0.90)
```

```
## [1] 22.37286 23.12928
```

This is alarmingly close to the CI for the mean. It suggests that it doesn't matter which one you use (and therefore that the CI for the mean would be better, since it makes better use of the data).

So, for your three points, you need to say something like one of the sets of three things below. If you think the skewness is a problem:

- right-skewed
- we need normality, or, the skewness will affect the mean too much
- use sign test to get a CI for the median instead

and if you think the skewness is not a problem:

- the sample size ($n = 202$) is large
- the Central Limit Theorem says that a sample this large will allow us to overcome the kind of skewness we see
- therefore we don't need to do anything.

A certain amount of mixing-and-matching is possible here, eg. "I see right-skewness; however, the sample size is large". I do not think it's fair to say that this histogram is itself symmetric, so you need somehow to say that the skewness we see is not critical, if you're going with the $t$ confidence interval.

Also, the Central Limit Theorem is not a cure-all: you cannot say that once the sample size gets beyond $n = 30$ (or whatever value you use for "large"), all your problems automatically go away. The way it works is that the larger the sample is, the less a certain amount of skewness matters, but if your skewness is really bad (eg. you have a lot of upper-tail outliers), you might need sample size $n = 1000$ or even bigger to be able to trust a $t$-test.

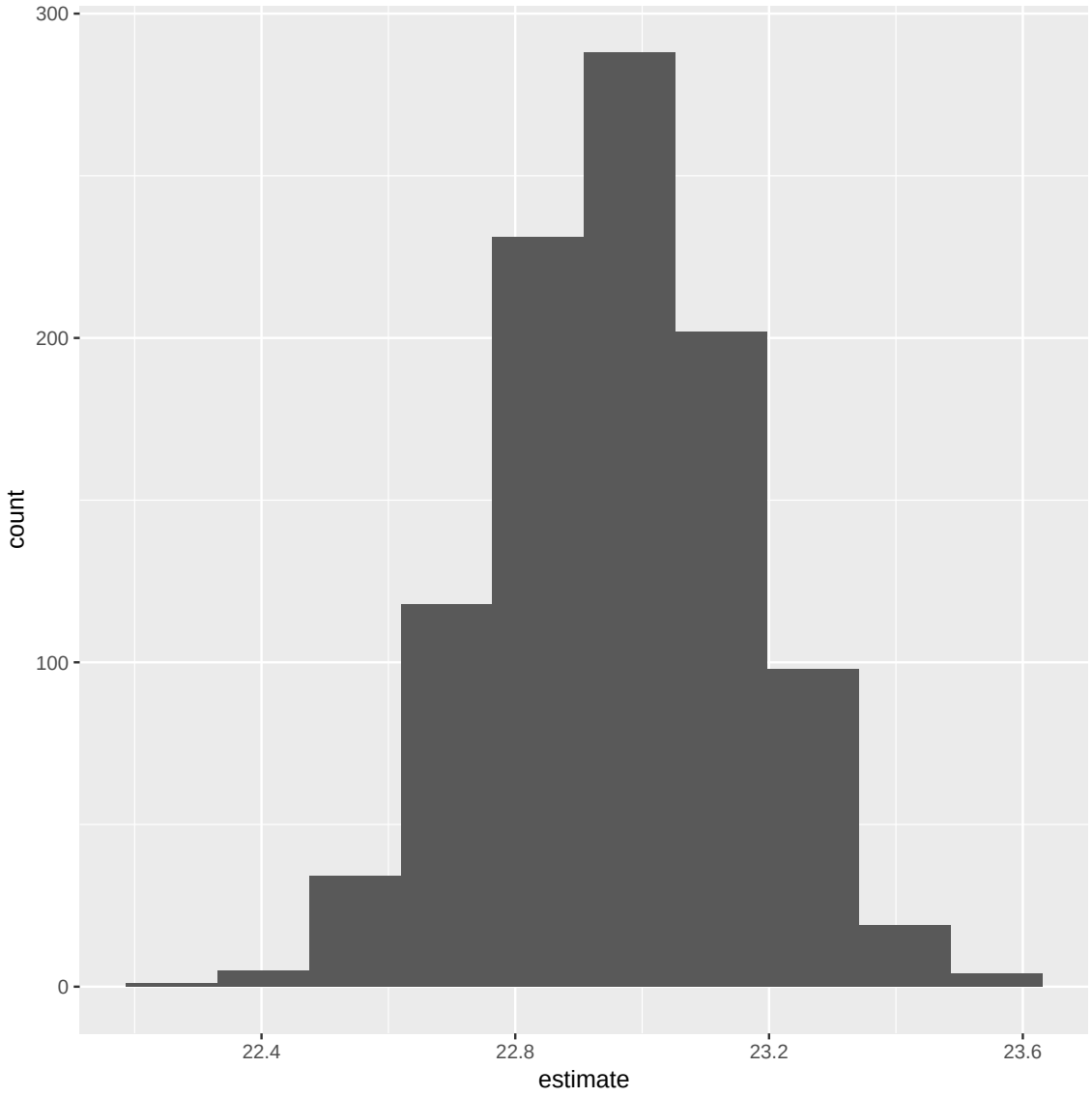
The crucial thing in all of this is the sampling distribution of the sample mean. This is what needs to be normal in shape. A way to assess this would be to do a bootstrap simulation, as I did in one of the assignment solutions. Here, my guess is that this will be pretty normal.

Borrowing from the journey-to-school problem on the assignment:

```
library(broom)
boots=athletes %>% bootstrap(1000) %>%
    do(tidy(t.test(.[["BMI"]])))
boots
```

```
## # A tibble: 1,000 x 9
## # Groups:   replicate [1,000]
##    replicate estimate statistic       p.value parameter conf.low conf.high
##        <int>    <dbl>     <dbl>         <dbl>     <dbl>    <dbl>     <dbl>
##  1         1 22.71426  126.4724 1.504892e-193       201 22.36012  23.06840
##  2         2 23.12807  122.4129 9.750376e-191       201 22.75552  23.50062
##  3         3 23.14104  101.3945 1.483052e-174       201 22.69101  23.59107
##  4         4 23.16050  106.4714 9.634209e-179       201 22.73157  23.58942
##  5         5 22.93688  116.2871 2.560587e-186       201 22.54795  23.32581
##  6         6 23.17302  126.2867 2.014584e-193       201 22.81120  23.53484
##  7         7 22.63629  123.1181 3.120380e-191       201 22.27375  22.99883
##  8         8 22.84822  115.8010 5.870757e-186       201 22.45916  23.23727
##  9         9 23.17970  126.6444 1.148992e-193       201 22.81880  23.54061
## 10        10 22.84738  129.5090 1.353206e-195       201 22.49951  23.19524
## # ... with 990 more rows, and 2 more variables: method <fctr>,
## #   alternative <fctr>
```

```
ggplot(boots,aes(x=estimate))+geom_histogram(bins=10)
```




and that looks about as normal as you could wish for.

In case you were wondering, I was doing a bootstrap $t$-test that the mean is *zero*, which of course makes no sense for a BMI, so I ignored anything to do with that test and focused on the "estimate", which is the mean of each bootstrap sample. If that distribution is normal, I'm good.

So the "right" answer was that we had a large enough sample size to overcome the skewness, but it was not necessarily reasonable to expect you to guess that with the resources you had on the exam, so I would also accept "skewed, therefore get a CI for the median". As ever, I was most interested in the thought process: could you make a logical argument to support your call, whatever it was? If so, I was happy.

5. This question is about power of hypothesis tests. Parts (a) and (b) refer to the same situation (described in part (a)), while parts (c) and (d) are independent of those and of each other.

   (a) (3 marks) A measurement varies according to a normal distribution with unknown mean $\mu$ but known SD 2. Suppose we are interested in testing the null hypothesis $H_0 : \mu = 8$ against the alternative $H_a : \mu > 8$. If in fact $\mu = 9$ and we have a sample size of 50, how likely are we to correctly reject $H_0$, at $\alpha = 0.05$? Give SAS code to do the calculation.

**Solution:**

This is `proc power`, thus:

```
proc power;
  onesamplemeans
  test=t
  mean=9
  nullmean=8
  sides=U
  stddev=2
  ntotal=50
  power=.;
```

The POWER Procedure
One-Sample t Test for Mean

Fixed Scenario Elements

| | |
|---|---|
| Distribution | Normal |
| Method | Exact |
| Number of Sides | U |
| Null Mean | 8 |
| Mean | 9 |
| Standard Deviation | 2 |
| Total Sample Size | 50 |
| Alpha | 0.05 |

Computed Power

| Power |
|---|
| 0.967 |

If you prefer, omit `nullmean` and replace `mean` with the difference between the null and the truth:

```
proc power;
  onesamplemeans
  test=t
  mean=1
  sides=U
  stddev=2
  ntotal=50
  power=.;
```

The POWER Procedure
One-Sample t Test for Mean

Fixed Scenario Elements

| | |
|---|---|
| Distribution | Normal |
| Method | Exact |
| Number of Sides | U |
| Mean | 1 |
| Standard Deviation | 2 |
| Total Sample Size | 50 |
| Null Mean | 0 |
| Alpha | 0.05 |

Computed Power

| Power |
|---|
| 0.967 |

The power is 0.967.

This is very high because we have a large sample size and small population SD; most of the time, the sample mean will come out very close to 9, and a population mean of 8 will be rejected.

Basic scheme: minus one per error, to a minimum of 1 if I thought you had made an "honest effort" (see below) with 2 or more errors.

`sides=1` also works. A very common error was to leave out `sides` entirely — how else is SAS to know that you are testing $H_a : \mu > 8$ rather than $H_a : \mu \neq 8$?

Choosing a two-sample test is a serious error. It's really only one error (the next `diff_satt` line is a knock-on from it and so is not really another error), but I couldn't in good conscience give you more than 1 for using the wrong test.

If you want any marks at all, you need to make an "honest effort" to get to the solution. `proc power` plus scraps will not get you any points. There were supposed to be seven things after the `proc power` line, eight if you went the `nullmean` way; two or three of them is not enough for a point.

(b) (2 marks) How would you change your code of the previous part to find the sample size needed to achieve probability at least 0.7 of correctly rejecting the null hypothesis?

**Solution:** Replace the missing `power` by the desired value 0.7 and replace the to-be-found sample size by "missing". For example:

```
proc power;
  onesamplemeans
```

```
test=t
mean=1
sides=U
stddev=2
ntotal=.
power=0.7;
```

I asked you how it changed, so all I want is the two changes. I don't need to see the whole thing again. In fact, if you wrote out the whole thing, I only checked those two lines.

Note that in SAS, if you want to leave something "unknown", you're actually leaving it "missing" rather than blank (that's what the dot does, since that's SAS's version of R's `NA`). Expect to lose a mark if you forgot the dot, probably in (a). If you did it in (b), I checked back to your (a), and if I had noted it there, I did not penalize you again in (b). The point of this part was for you to say how you would change what you did in (a), and if you could express that somehow, I was happy.

This was a pretty easy two points. Even if you had no idea how to do (a), you can still get two points in (b) with a sufficiently clear description of what you would do here, had you been able to do (a).

The POWER Procedure
One-Sample t Test for Mean

Fixed Scenario Elements

| | |
|---|---|
| Distribution | Normal |
| Method | Exact |
| Number of Sides | U |
| Mean | 1 |
| Standard Deviation | 2 |
| Nominal Power | 0.7 |
| Null Mean | 0 |
| Alpha | 0.05 |

Computed N Total

| Actual Power | N Total |
|---|---|
| 0.715 | 21 |

This is backwards from the usual way around: the power we got above was higher than we needed here, so our previous sample size was "too big" and we could get the accuracy (power) we wanted by taking a smaller sample size, 21 rather than 50.

(c) (3 marks) We are now comparing a treatment against a control. The standard deviation of both sets of measures is 10. Suppose we take a sample of 25 observations from each group and test the null hypothesis that the treatment and control have the sample mean, against an alternative that the treatment mean is greater. If in actual fact the treatment mean is 5 units higher than the control mean, what R code would *calculate* the power here?

**Solution:** The word *calculate* means "exactly", which rules out doing a simulation, so we need to fire up `power.t.test`:

```
power.t.test(delta=5,sd=10,n=25,type="two.sample",alternative="one.sided")

##
##      Two-sample t test power calculation
##
##               n = 25
##           delta = 5
##              sd = 10
##       sig.level = 0.05
##           power = 0.5390023
##     alternative = one.sided
##
## NOTE: n is number in *each* group
```

The power turns out to be 0.54.

This is actually doing a Welch-Satterthwaite unequal-spreads test (which is R's default, and that is fine). `power.t.test` seems not to have the option to get power for a pooled $t$-test. I thought you'd be able to stick `var.equal=T` in there and get it, but no.

There are some things to get right: `delta` is difference between null and true mean (differences), `sd` is the population SD of each group (assumed known and equal), `n` is the sample size in *each* group (unlike SAS, which uses the two sample sizes added together), `type` gives the type of test (one-sample or two-sample) and `alternative` says whether it's one-sided or two-sided. For power calculations, it doesn't matter *which* one side it is (greater or less), so this works (and, say, "greater" doesn't). There is no equivalent to SAS's `test=t`, because R's function works *only* for $t$-tests, not for things like ANOVA or tests for proportions (SAS's `proc power` also does both of those).

This is another one of those cases where you need clear enough thinking to distinguish one-*sample* from one-*sided*.

I said that something like `alternative="greater"` doesn't work:

```
power.t.test(delta=5,sd=10,n=25,type="two.sample",alternative="greater")

## Error in match.arg(alternative):  'arg' should be one of "two.sided", "one.sided"
```

I decided to let that go for two reasons: one, it shows the right intention, and two, if you were to try it in real life and get that error message, it would be pretty clear to you what to do next.

Same marking scale: minus one per error, down to a minimum of 1 (for which you needed to write `power.t.test()` and fill in the right inputs, even if you didn't get any of their values right). If you generated combinations of values of `n` and/or `delta` and then fed those into `power.t.test`, you probably got the points (if your combinations included the right values), but you may not be so lucky next time. Why make extra work for yourself (and me)?

I clearly wrote "R code", so if you wrote `proc power` code, I gave you zero and moved on to the next paper. So if that's what you did, it was a complete waste of your time.

(d) (3 marks) Look at Figure 8. Explain briefly what the results at the bottom indicate. Do *not* explain what each line of the function does. You may find it helpful to make *one* comment about what the function *as a whole* does, one comment about what the lines of code below the function are doing, and one comment about the results at the bottom.

Some R code that you may have forgotten: `rep("x",n)` repeats the text `x` $n$ times, `c()` glues two vectors together, for example:

```
c(1:3,8:10)
```

```
## [1]  1  2  3  8  9 10
```

and `tibble` creates a data frame out of the variables fed into it.

**Solution:** This is power by simulation, but in an unfamiliar guise. Here are the three points that you can make:

1. The function generates two random normal samples (with the input means, SDs and sample sizes), does a (two-sided) two-sample $t$-test and returns the P-value.
2. The code below the function repeats the $t$-test 1000 times, and counts how many P-values were less than 0.05 (rejecting) or not (not).
3. The null hypothesis is wrong (the input means to the function for the two samples are different), so the power of the $t$-test to correctly reject the null is $457/1000 = 0.457$, when the two samples are from populations with means 60 and 55 and sample sizes 25 and 22.

If you can get to the third point successfully without making the first two, I'm happy with that. But, the usual thing: if you go astray in your answer to the third point but you've made some valid points along the way, expect to get credit for them. The first two things are "scaffolding" intended to help you get to the third one, which is what the marks are for.

In this simulation, group `a` has mean 60, SD 8 and sample size 25, and group `b` has mean 55, SD 10 and sample size 22. These two groups have different (population) means, and so we are simulating the power of the two-sample $t$-test to declare a difference when there actually is one. The table at the bottom says that the simulated (estimated) power is 0.457.

The power isn't very big because the sample sizes are not very big (or the population SDs are relatively large).

Mark scale:

- 3 if you tell me all of what's in the second-to-last paragraph including the numbers.
- 2 if you convince me that you know we're estimating the power of a test, and that the estimated power is 0.457
- 1 if you seem to have some clue about what the code is doing, even if you don't know what it's for.
- 0 otherwise.

My dividing line between 1 and 2 is whether or not you knew what the *purpose* of the code was. It's one thing to know what I am doing, but another to know *why* I am doing it and what the results actually *mean*. The latter is a "higher-order skill" and so deserves to be worth more.

This simulation is a little more flexible than `power.t.test`, because it allows the groups to have different SDs and different sample sizes. I wrote the function to accept any means, SDs and sample sizes for maximum flexibility. For example, if I make the population SDs smaller, leaving everything else the same, I should increase the power (I have to repeat the function for annoying technical reasons):

```
samp=function(true_1,true_2,sd_1,sd_2,n_1,n_2) {
    r1=rnorm(n_1,true_1,sd_1)
    r2=rnorm(n_2,true_2,sd_2)
    g1=rep("a",n_1)
    g2=rep("b",n_2)
    d=tibble(value=c(r1,r2),group=c(g1,g2))
    ans=t.test(value~group,data=d)
    ans$p.value
}
pp=replicate(1000,samp(60,55,4,5,25,22))
tibble(pp) %>% count(pp<=0.05)
```

```
## # A tibble: 2 x 2
##   `pp <= 0.05`     n
##          <lgl> <int>
## 1        FALSE    37
## 2         TRUE   963
```

with smaller population SDs, the means are in effect further apart, so we should be more easily able to reject the null that the means are equal. And so it proves; the power is up over 0.95.

6. The typical age of onset of diabetes (that is, the age at which someone with diabetes is diagnosed as having diabetes) is 37 years. A sample of people with diabetes is taken from a certain population. The data are shown in Figure 9. We are interested in whether the typical age of onset for these people is different from that of other people with diabetes.

   (a) (2 marks) A histogram of the ages of onset is shown in Figure 10. Why do you think the statistician in this study chose to use a sign test rather than a $t$-test?

**Solution:** First thing to note is probably that the data are (slightly) right-skewed. Why does this matter? The second thing to note is either (i) the $t$-test assumes normality and so is no good here, or (ii) the median is a better measure of centre than the mean, and the sign test is a test for the median.

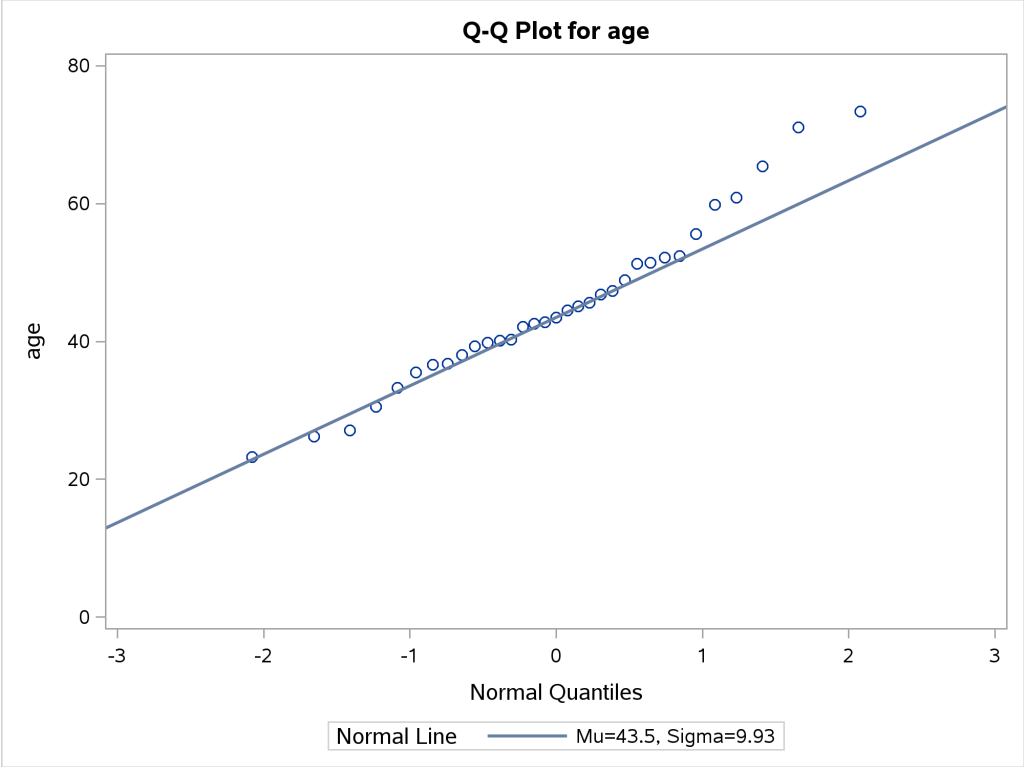
There are two points you need to make: something about the distributional shape, and something about why it matters.

With 33 observations, you might argue that the central limit theorem comes into play and the distribution is "normal enough", but that's not the call the statistician made. If you wish to make that point instead, go ahead. If it had been me, I would have said that this distribution was normal enough, and gone ahead and done a $t$-test. But then we wouldn't have had a sign test question. (It's like when something implausible happens early in a TV show, and you realize later that it had to happen that way or else there would have been no story to tell and thus no TV show.)

Some people said that a different plot might have been better. I think a normal quantile plot would have been the best thing:

```
proc import
  datafile='/home/ken/diabetes.txt'
  out=diabetes
  dbms=dlm
  replace;
  getnames=yes;
  delimiter=' ';

proc means median qrange;
  var age;

proc univariate noprint;
  qqplot age / normal(mu=43.5 sigma=9.93);
```

```
                    The MEANS Procedure

                  Analysis Variable : age

                                    Quartile
                       Median           Range
                ----------------------------
                  43.5000000      13.4000000
                ----------------------------
```




I think those high-end points are a bit too far above the line to call it normal, and therefore there is a little skewness (or upper-end outliers). But you could almost make the call the other way, even then, since they are not *far* above the line. You can see how close a call this is, especially given that we don't need *exact* normality for the $t$-test anyway, merely approximate normality, and a sample size of $n = 33$ allows the Central Limit Theorem to help us a little, which might be all we need.

By the way, those values for `mu` and `sigma` are the median, and the IQR divided by 1.35. (That's why I calculated median and IQR above the plot.) This makes the points at the upper end look a bit further off the line than for `mu=est` and `sigma=est`. Those points at the upper end are inflating the mean and SD a little.

(b) (2 marks) Write appropriate null and alternative hypotheses for the sign test.

**Solution:** Null: (population) median age is equal to 37; alternative: (population) median age is not equal to 37 (using "different" as a clue that a two-sided test is called for). That is to say, the median of the population from which these 33 individuals were sampled either is or is not 37. *Define your symbols.* $M$, or more especially $\mu$, does not mean "population median" until you tell me that it does.

Ways to lose a mark here:

- Using a symbol like $\mu$ or $M$ without telling me what it means (implicitly defining it by restating the hypotheses in words is fine).
- Not using a symbol at all (writing something like $H_0 = 37$ — *what* is it that is supposed to equal 37?)
- Using < or > in the alternative hypothesis.

- Using the word "sample" in your hypotheses. Hypotheses are statements about a *population* parameter; the sample median either is or is not equal to some specified value, and you can check such a "hypothesis" with certainty by looking at the data. The reason a test of hypothesis is scientifically interesting is that we *don't know* the population median and can only ever estimate its value.
- Using the word "mean" in your hypotheses. The sign test is a test for the population *median*. Some people got this wrong despite (apparently) an excellent discussion in (a) about how the sign test was the right thing because it tested the median. (This might be the same breed of sloppiness involved in claiming that boxplots enable you to compare means. Boxplots don't have means, unless they are SAS ones where the mean is marked by a diamond. Tukey's original conception of the boxplot was based on medians and "hinges", that is, quartiles.) Some people used words here like "typical value", which I let them get away with even though it is sloppy.

It was hard to write something and get zero. You either had to write down things that were not hypotheses, or you had to make two serious mistakes. (You would still get 1 if you made one of the errors above along with something else non-fatal.)

(c) (2 marks) A sign test was carried out in SAS, as shown in Figure 11. Obtain a P-value from the output, and state your conclusion in the context of the data.

**Solution:** The P-value for the sign test is 0.0046. We conclude that the median age of onset of diabetes is not 37, since it is smaller than 0.05 (or 0.01 or whatever you are likely to have used).

If you thought the alternative hypothesis needed to be one-sided above, you should be consistent here. Check that you are on the correct side, and if so, divide 0.0046 by two. The cases are, letting $M$ be the population median: if you have $H_a : M > 37$, you are on the correct side (the majority of the data values are above 37) and the P-value is 0.0023. If you have $H_a : M < 37$, you are on the wrong side, the P-value is "large" and you do not reject the null median of 37 in favour of less. (The P-value in this case is actually $1 - 0.0046/2$, but these wrong-sided ones are always large and cannot lead to a rejection.)

There is some handy phrasing in the last sentence of the question preamble: "the typical age of onset for these people is different from that of other people with diabetes", which will do as a conclusion, though it's more precise to talk about the median, since that is what is being tested.

One point for 0.0046, and one point for a statement about median age of onset of diabetes. Feel free to get to the second point by starting off with "reject the null hypothesis", but you must make some kind of statement about median (or "typical") age of onset of diabetes in order to get the second point.

"The P-value is less than 0.05" as an opening to your answer is not enough, since I want to know what the P-value *is*: there are three in the output and I want to be sure that you have the right one. Of course, using a sentence like this to start the *second* part is fine. Something like this is ideal:

> The P-value is 0.0046. This is less than 0.05, so we reject the null hypothesis and conclude that the median age of onset of diabetes is not 37.

I wanted you to finish with an actual statement of what you concluded. Don't just give the null that you're rejecting and leave it at that. Think of your audience as looking for the TL;DR and make it easy to figure out what that is.

As ever in this kind of situation, make sure your English is clear enough to distinguish your *hypothesis* from your *conclusion*. Something like this is good:

> We reject the null hypothesis that the median age of onset is 37 and conclude that the median age of onset is in fact not 37.

This separates the statement of the hypothesis (the bit after the first "that") and the statement of the conclusion (after "conclude that"), so that it is clear which is which. The next one is *not* clear; in fact it's wrong:

We reject the null hypothesis that the median age of onset is not 37.

The bit after "that" is, from the point of view of the English, the statement of the *hypothesis*, whereas the writer seems to mean it to be the statement of the *conclusion*. Replace "that" by "and conclude that", and all is good. It's actually easier *not* to state the null hypothesis here, at least not in this sentence (you can state it in the previous sentence), since that makes the answer unwieldy and long, and makes it easier for you to get your language mixed up. Look back at the hypotheses you wrote earlier and use the appropriate one to help you write something that looks like a conclusion.

(d) (3 marks) Some R output for the same data is shown in Figure 12. Use this output to describe how SAS obtained its P-value.

**Solution:** The sign test requires us first to count the number of data values below and above 37 (the null median). The output shows that there are 8 below and 25 above. (There are actually no values exactly equal to 37, as you can check from Figure 9.) Then, to get a P-value, we need the binomial distribution with $n = 33$ (33 data values) and $p = 0.5$. This is the data frame at the bottom of Figure 12. The P-value is the probability of 25 successes or more (or of 8 successes or less, but that isn't shown) doubled, because the test is two-sided. So what happened is that the probabilities from 25 to 33 were added up, and then doubled:

```
sum(dbinom(25:33,33,0.5))*2
```

```
## [1] 0.004551384
```

and that's the same P-value that SAS got for the sign test.

You need to say two things:

1. that the data frame at the bottom of Figure 12 is a table of the binomial distribution with $n = 33, p = 0.5$;
2. in that table, the P-value is the sum of the probabilities from 25 to 33, doubled.

If you are confident enough that this is what was done, you don't need to actually add the values up, for example with your calculator, but it's smart as a check that you were right. The probabilities out in the tail of the distribution are pretty small, so, in order to reproduce the answer SAS got, you might be able to get away with adding up not very many of them. Let's just keep 5 decimals:

$$\begin{aligned} P(25) &= 0.00162 \\ P(25-26) &= 0.00162 + 0.00050 = 0.00212 \\ P(25-27) &= 0.00162 + 0.00050 + 0.00013 = 0.00225 \\ P(25-28) &= 0.00162 + 0.00050 + 0.00013 + 0.00003 = 0.00228 \end{aligned}$$

Twice that is 0.00456, which is very close to the exact answer (calculated by R). This is enough to show how it was done. Also enough is to produce code that shows how it was done, in a way that can be checked from my output (the `sum(dbinom())` thing from above).

I think a good way to tackle this is to think about how you would do it sitting in front of R Studio (without `smmr`). What's the first thing you would do? Count up the values above and below 37. That

is the first thing done in Figure 12, which is encouraging. There are 25 above and 8 below. (One point for getting this far.)

Next up, the P-value. You probably remember that this comes from a binomial distribution with $n$ equal to the sample size and $p = 0.5$. At least some of that binomial distribution is in the tibble at the bottom of Figure 12. You might even remember that we had to add up some of those binomial probabilities, and possibly even, because this test is two-sided, that after we're done adding we'll have to multiply by two. (A second point for getting this far.)

A lot of people thought that you just take those probabilities that I gave you, from 20 up to the top, and add them all up. But why 20? I wanted to give you at least as many probabilities as you needed, but that can and here does include some extra ones. Can you think clearly enough to see what is needed to get the P-value? It is the probability of "the value observed or a more extreme value". That would mean 8 or fewer successes (not in the table I gave you) or 25 or more successes (which *is* in the table I gave you). So take the probabilities from the observed 25 (number of patients with age of onset greater than 37) to 33. Add them up, and then multiply by 2 since the test is two-sided. Get all of that, and you have your third point. If you have time, you can check the addition on your calculator.

The boundary between 1 and 2 points was somewhat flexible. I was also swayed by whether you seemed to have made "substantial" progress towards the answer (for which I gave 2 points) or whether you seemed to have only just started out (1 point).

7. Trace metals in drinking water affect the flavour, and an unusually high concentration of them can pose a health hazard. Zinc concentration in water at the bottom of a lake and in water at the surface was measured at each of 10 different locations. We are interested in whether zinc concentration is higher on average at the bottom of the lake. If it is, drinking water should be drawn from near the surface of the lake. The data are shown in Figure 13.

   (a) (2 marks) Explain briefly why a matched-pairs analysis would be more suitable here than a two-sample analysis.

**Solution:** Make some point about the data being paired up: for example, there are two paired-up measurements for each location, one on the surface and one at the bottom of the lake. Or, each location gives you two measurements. Or, there are 20 data points altogether but only 10 locations. Something like that. The word "location" needs to be there.

People found this harder than I expected. A lot of people thought that it was the same *lake* that made it matched pairs. Think about what the experimental units are: locations *within* a lake. There are two measurements of zinc (the response) at each location, the lake bottom at that location and the lake surface at that location. To see that it can't be the lake that makes it work, note that what could be done is that we use the same 10 locations, but now flip a coin to decide whether we take a bottom measurement or a surface measurement at that location (we only take one). This makes it two independent samples: we end up with some surface measurements and some bottom measurements, all at different locations.

Talking about the layout of the data doesn't get there. Someone could have erroneously recorded the data this way "to save space" even if there were actually 20 different locations, each one of which was *either* surface or bottom (that is to say, a two-sample experiment). Row 6 would then tell you that you were looking at location 6 *out of the bottom measurements* and location 6 *out of the surface measurements*, which might be different physical locations. The decision about paired vs. two-sample has to come from the *design* of the experiment, which will be found in the description of the data: usually in the question on an exam, but in real life you might have to ask the experimenter what they did, so that you will do the right analysis.

I am reminded of something I also saw earlier in this exam: some people seem to know the theory ("two measurements on the same subject"), but seem not to be able to demonstrate how the theory applies to the situation described, that is to answer "what are the subjects here?" (locations) and "what are the two measurements?" (measurements of zinc taken at the surface and the bottom). This is an applied course; knowing the theory is one thing, but that is nearly *worthless* if you cannot apply it to the situation in front of you.

This is rather like answering "reject the null hypothesis" to a question that says "what do you conclude, in the context of the data?". It shows that you know something about what to do, but not why you're doing it or what it's for. (Look up "Bloom's Taxonomy": we need to be beyond Remembering and Understanding, since we are doing Applying and Analyzing, and possibly Evaluating, here.)

   (b) (3 marks) Figure 14 and Figure 15 show two possible analyses for these data. Which of the two analyses is more suitable? What, therefore, do you conclude in the context of the data?

**Solution:** Figure 15 contains `paired=T`, so it is the appropriate analysis. The other one is a two-sample analysis, which we already ruled out. We wanted a matched-pairs test, which is what we have. (I didn't ask for a reason, so I didn't penalize you if I disagreed with the reason you gave.)

The null hypothesis is that the mean difference is zero, with the alternative that the mean difference is greater than zero (one-sided). Or, the alternative says that the mean zinc concentration at the bottom is higher than at the surface.

Question 7 continues... This page: _____ of possible 5 points.

The P-value for the matched-pairs analysis is 0.0004, very small, so we reject the null in favour of the alternative. That is, the mean zinc concentration at the bottom really *is* higher than at the surface.

If you thought the two-sample analysis was better, you should be consistent and interpret its non-significant P-value correctly. This was worth 1 if you did it right.

I was hoping you would name a figure, 14 or 15, but if you had the right P-value that was good too.

Make sure you get the right hypotheses. The alternative translates to the mean zinc concentration being higher at the bottom of the lake. Also, "in the context of the data" means that you should finish with a statement about concentrations of zinc, not "reject the null hypothesis" or "the mean difference is greater than zero". I would let you get away with the latter if you said somewhere which way around the differences were.

We *never* choose a test on the basis of its P-value. This is called "P-value hacking" and leads to a lack of respect for statistics among people who work outside it. So don't do it. We as statisticians might compare P-values as a way of learning about why tests behave in different ways, but *we do not choose a test because we like the conclusion it leads to.* There has to be a reason based on the design of the experiment (as here) or on the nature of the data (in choosing between $t$ and sign) that leads to choosing one test rather than the other *without* looking at its P-value.

It's all right to look at both P-values *after* you have decided which test you like, as a way of understanding what is going on. In this case, there is a lot of variation from one location to another, which (correctly) gets accounted for in the matched-pairs test, and (incorrectly) does not in the two-sample test. This is the explanation for the big difference in P-values.

(c) (3 marks) What specific assumption are we making in order to trust the results of our $t$-test? Which one of Figure 16 and Figure 17 enables us to assess this assumption? What do you conclude? Explain briefly.

**Solution:** We need the *differences* to be approximately normal. The distribution of the actual measurements doesn't have to be even approximately normal (because the test is based on the differences). So we need to look at Figure 17 rather than Figure 16.

Make a call about whether you think this is normal enough or not. If you think not, you ought to be able to name something that stops the distribution being sufficiently normal for your tastes. You could (just about) make the case for an outlier at the top (and maybe the lowest value not being low enough). But I think this is not really enough of a departure from normality to be worth worrying about. So I would trust the matched pairs test. Another consideration is that the P-value is so small that even if normality is off by a bit (however you measure that), the "right" P-value is still going to be small.

I added the word "specific" to the question to offer you a hint that "normality" is not a complete answer; I want to know precisely *what* it is that has to be (approximately) normal. If you thought the two-sample analysis was the better one, then you should again be consistent and ask whether the *two* normal quantile plots in Figure 16 are *both* approximately normal. Having said that, if you managed to get to preferring Figure 17 and talk *somewhere* about differences (being the important thing to look at), then I was happy. Quite a lot of people asserted that it was "the data" that had to be normal (too vague), but went on and semi-changed their mind to say that Figure 17 was of the differences, which was the what had to be normal since this was matched pairs. Getting there in the end was what I was looking for, and I was willing to forgive you a misstep along the way if your final conclusion was sound.

If you decided that the other Figure was the one, you were fighting for one point (if you make a sensible deduction from what that Figure has to say).

Once again, there was a bit of flexibility about what earned 1 and what earned 2 points. If you said

"Figure 17" and not much else that would get you 1, but if you wanted 2 you had to mention that the differences were the important thing, or make a sensible deduction from the normal quantile plot. As before, my mental guideline was "did you get most of the way to the right answer?"

I pretty much didn't care *what* you concluded from the normal quantile plot, as long as you supported it by something!

8. A new approach to prenatal care is proposed for pregnant women living in a rural community. The new program involves in-home visits during the course of pregnancy. A trial with 15 pregnant women is designed to evaluate whether women who participate in the new program deliver healthier babies than women receiving the usual care. The outcome is the APGAR score indicator measured 5 minutes after birth. APGAR scores range from 0 to 10 with scores of 7 or higher considered normal (healthy), 4–6 low and 0–3 critically low.

   The data are shown in Figure 18. The column `care` shows whether each pregnant woman received the usual care (`usual`) or extra prenatal home visits (`visits`).

   (a) (3 marks) Figure 19 shows the results of a hypothesis test run on these data. The first part of the output from `median_test` shows a table. What do the numbers 5 and 1 on the second row of that table represent? Explain briefly.

**Solution:** This row of the table tells you how many of the women who received extra prenatal visits had babies whose APGAR scores were above the overall median and below it.

The top of Figure 19 tells you that the overall median APGAR score was 7. Thus, 5 of the women who received extra prenatal visits had babies whose APGAR score was above 7, and only one of them had a baby whose APGAR score was below 7. (There was one woman in this group whose baby had an APGAR score of exactly 7, which is omitted from this table. Likewise, there were actually 8 women in the `usual` group, two of whose babies had an APGAR score of exactly 7 and were omitted from this table.)

If you want the third mark, you'll need to say that the 5 and 1 refer to scores above and below 7, not just above and below the overall median. Correctly using that value 7 obtained at the top of Figure 19 shows the proper understanding.

There were three points that needed to be made for all three marks:

- we were *only* considering the women who received extra prenatal visits. (I wanted you to make it clear which women we were talking about, as if you were trying to sell doctors on this new initiative. "Visits group" didn't quite do that for me; "women in the new program" would be (just about) clear enough.
- the overall/grand median APGAR score was 7. (Not the "null median" or the "population median": this was the median of all the APGAR scores in the data set.)
- Of the women receiving extra prenatal visits, 5 of them had babies with APGAR score over 7 and only one of them had a baby with APGAR score below 7.

There were different ways to get 2 marks: you could make all of those points, but fail to make one of them clearly enough (not specifying precisely which group of women we were talking about, for example). Or maybe you made only two of those points, as in for example comparing APGAR scores with "the median".

In a common theme for 3-point questions, I tried to give you 2 if you got more than halfway, and 1 if you seemed not to have gotten that far. It was possible to make two mistakes and still get 2, if the rest of your answers were good enough.

   (b) (2 marks) In Figure 19, look again at the table in the first part of the output from `median_test`. Does this suggest that the new program is helpful? How can you tell? Explain briefly. (This part is not asking about P-values.)

**Solution:** If the new program is successful, the women receiving the extra visits should have babies whose APGAR scores are higher on average than those of the women who received the usual care. This

seems to have happened: 5 of the 6 women who received the extra visits had babies with above-average APGAR scores, while only 2 of the 6 women who received the usual care had babies with above-average APGAR scores. (Or anything equivalent to that, like "women receiving extra visits tended to have babies with above-average scores, while women receiving the usual care tended to have babies with below-average scores".)

In this part, I'm testing your intuition about what kind of result you would expect from the hypothesis test. I'm not asking about the P-value and its interpretation, yet.

I was more relaxed about what I would accept in this one (compared to (a)). If you did some kind of comparison of APGAR scores between the two groups and concluded that the scores were better for the extra-visits group, I was happy. Or, you could even conclude that there was "approximately an even split" of values above and below in the two groups. I happen not to agree with that, but as long as you were logically consistent (ie. you said there was "not much difference" after that) I would go with it.

I tried to give you 2 if you addressed the right kind of issues. If you only got 1, it's because I felt that your explanation was not clear enough or didn't really get at the issues.

Two ways to get 0: (i) to fail to compare the two groups of women in any way, (ii) to guess an answer but provide no explanation. If I ask for an explanation and you don't give me one, don't expect to get anything in return!

(c) (3 marks) What do you conclude from the result of the test at the bottom of Figure 19, in the context of the data? In writing your conclusion, bear in mind what the researchers running this study are trying to prove.

**Solution:** The last sentence is intended to make you look back at the question. The researchers are trying to see whether the extra visits make the women "deliver healthier babies", that is, with a *higher* APGAR score. That is to say, we should be doing a *one*-sided test. This output does a *two*-sided test, and so we need to (i) check that we are on the correct side, and (ii) if we are, divide the P-value by 2. The "correct side" thing is what we did in the previous part; most of the women who had extra visits *did* deliver a baby with an above-average APGAR score (and those who did not, did not). Thus our P-value should be half of 0.0790, that is, 0.0395. That means that we can reject the null hypothesis that the new program has no effect and conclude that it has a *positive* effect on APGAR scores, as the researchers were hoping to show: women who receive extra prenatal visits really do have babies with higher APGAR scores.

Three points if you got all of that. If you halved the P-value, I checked carefully for evidence that you had done so because you wanted a one-sided test, even if you didn't explicitly say so.

If you thought that a two-sided test was appropriate, you need to say that the P-value is 0.0790 and that we cannot reject the null hypothesis (at $\alpha = 0.05$); therefore we do *not* have evidence that the new program has an effect on APGAR scores (or that the difference observed could be attributed to chance). That is to say, as far as we can tell, the new program and the usual one are equally good.

Two points if you went this way and got it all right. A surprising number of people got their null and alternative hypotheses messed up and thus came out with the wrong conclusion: the null hypothesis is that the median APGAR scores on the new and usual programs are *equal* (or that there is no association between group and being above/below median). Strictly, the alternative that goes with P-value 0.0790 is that there is a *difference* between the medians (that one of the programs is better than the other, but the "visits" one could be *worse*). If, for example, you chose $\alpha = 0.10$ and thus rejected the null of no difference, this is what you needed to conclude: "there is a difference". I saw again some confusion between the statement of hypothesis and conclusion. Look back again at the discussion in the sign test question if this happened to you. "I conclude that" is a handy way to make it clear.

Anyhow, one point covers everything from quoting me the P-value to trying to make a (two-sided) conclusion and messing it up.

If you did a two-sided test, I appreciated seeing some kind of consideration of the apparent inconsistency with the previous part: that the result would be significant at $\alpha = 0.10$, or that the result could have failed to be significant because of the small sample size, something like that.

This is, in retrospect, a very small sample. It was described, where I got the data from, as a "pilot study" whose intention was to demonstrate that there is some potential to the new program. Since this pilot study was successful in that regard, the researchers can apply for funding for a bigger study, maybe with 100 or 200 pregnant women (the actual sample size might be based on a power study of some kind). If an effect of the size seen here is repeated in the bigger study, it will be strongly significant.

The other thing I was thinking about was that APGAR scores are limited to the interval 0–10, so there ought not to be outliers or very non-normal distributions. That said, if you look back at the data in Figure 18, the scores of 2 and 3 in the `usual` group do look much lower than the rest, because most of the scores are bunched up at the top, 7 or higher. So maybe the distribution *is* left-skewed, and therefore using Mood's median test was indeed a smart idea.

If you want to know more about the APGAR score, here is a nice chart:

https://www.abclawcenters.com/wp-content/uploads/2016/06/Apgar-Scoring-System-Diagnosing-Birth-Injuries.jpg.

It's done within 5 minutes of the baby being born; the baby is given a score of 0, 1 or 2 on each of the categories shown, to determine whether the baby is healthy or needs more medical attention.