Presentation of Tables, Graphs and Maps

Alex Thomson

2021-07-14

Contents

Chapter 1

Introduction

Visualising data is an essential part of communicating messages and results to any form of audience. An ineffective visualisation of data can communicate a very misleading message.

Building skills in data visualisation can help you to understand and see important results in other people's tables, graphs, and maps. This is in addition to enabling you to create informative visualisations of your own.

The aim of this document is to provide comprehensive guidance on the presentation of data in tables, graphs, and maps. This will include both general guidance and more specific advice on different types of visualisations. We intend to provide some principles of good graphical, tabular, and cartographic practice. By providing this advice, we hope to assist anyone in their future work, especially when it comes to the writing up of research results for an audience.

This guide is intended for anyone who wishes to develop their data visualisation and reporting skills. The advice presented here will be applicable to a wide variety of situations and is not specific to certain topics. Additionally, we hope that users of all ability levels will be able to take this advice to mind in their future projects and their everyday interactions with data.

This resource will start by exploring some general guidance on the presentation of data before going into more specific detail on the use of tables, graphs and maps (an increasingly popular method of presenting data). It then provides advice on ensuring your visualisations are accessible, with consideration on the use of colour.

Chapter 2

General Guidance

Through developing your data visualisation skills, you can generate a wide variety of graphical/cartographic/tabular representations of data. This could be anything from simple bar graphs and line graphs to complicated cartograms. However, regardless of the complexity of your chosen data visualisation technique, there are certain principles that should always be followed:

2.1 Effectiveness

By effectiveness, we mean you should be ensuring that you are using the right type of visualisation for your objectives and priorities. This is the first crucial step in making sure what you produce is effective at displaying the message you intend to show. If you pick the wrong method, your visualisation will not be effective regardless of its quality.

Maps are of course for displaying data which have some form of geographic component.

Tables are suited for presenting structured numerical information; consider tables of means across some groups, frequencies, or some statistical information. This makes them ideal for when the message is in the specific numbers and potentially the relationship between them.

Graphs are quite multi-purpose; there is a type of graph for almost any message you could be wanting to convey. In general, we would choose to use them for indicating trends, making broad comparisons, or showing relationships.

2.2 Readability

All elements of your visualisation should be legible, understandable, and coherent. In a word, readable. While this largely relates to any textual elements of your visualisations, the principle is applicable to the whole visualisation.

This includes having titles and headings which concisely explain the content. It should be informative without being overly long and confusing. The same goes for any further labels such as axis labels for graphs, column headings for tables and geographic labels on a map.

Details to consider mentioning include measurement units, geographical coverage, time, the source of the data and any relevant statistics. Of course, the elements of a visualisation will vary depending on what visual you produce, but they should always be easy to read and understand. You can achieve this by avoiding language beyond the scope of your target audience, providing the necessary information needed to read your visual and presenting the element in a simple and tidy manner. You will find further guidance on specific elements in each of the subsequent sections of this guide.

2.3 Tidiness

A visualisation should never be cluttered. This follows on from readability, although more specifically relates to positioning and spacing of elements as well as avoiding using unnecessary elements.

This includes making sure no elements are overlapping; there should be adequate spacing between them without there being so much that it makes the visualisation look empty. This can also be described as making good use of "white space".

There is much more to be said on this topic, but these are mostly specific to the type of visualisation you are using. The general principle of ensuring your visual is neat and organised is always applicable.

2.4 Accessibility

Accessibility has become an increasingly important aspect of data presentation in recent years. Ensuring good practice in accessibility will help in getting an even wider audience to see our research and use our results. The Government Statistical Service makes content accessible to those with impairments to their vision, hearing, mobility, and thinking/understanding skills. For our purposes, we are mostly concerned with visual impairments.

There are some general principles on accessibility, including making sure you explain any uncommon abbreviations, avoiding clutter and keeping information concise. However, we are focusing on the use of colour. Further guidance on this is included in the accessibility section of this document. This includes considerations of colour blindness, cultural context, and the use of saturation/hue/luminance.

2.5 Consistency

This is mostly relevant when you are intending to use multiple visualisations across your report. When doing so, it is important to ensure you maintain a level of internal consistency.

This involves many aspects. For instance, if you intend to disaggregate your visuals by the levels of a variable, pay attention to the order you put these categories in. They should be kept to a logical or ascending/descending order and this order should be kept the same for the sake of consistency and readability.

The same goes for when using colours to indicate certain characteristics of the data; keep the meaning of the colours consistent.

Of course, this is also important for all the smaller details such as the font, size and face (bold/italic) of text. In essence, try to keep the formatting between visualisations as similar as possible. Generate your personal visual style and stick to it. Changing things up too much will just confuse your audience and reduce your visual's readability.

2.6 Informative

A good data visualisation serves to succinctly show a message about our findings. We aim to inform our reader. Usually, it would be accompanied by some text which helps to interpret the visualisation, placing it into a wider context or providing more formal details such as the results of a relevant statistical analysis.

However, a good data visualisation should be self-explanatory and should be able to serve as a stand-alone piece. The reader should be able to understand the message without constantly referring to the text. Much of this can be accomplished by sticking to the particulars of keeping your visual tidy and readable.

Whenever creating a table, graph, or a map, you should include the source of the information from which the visualisation was created. This aids the credibility of your visualisation but also ensures a properly informed audience. An exception is when all information that is used for visualisations in a report comes from the same source. In this case, you should clearly indicate the source in advance of

your visualisations. This also means making sure that your visual is necessary in the first place. Consider the following: Can you achieve the same message with some simple text? Can a visualisation accurately demonstrate your results, or would it be distracting? Are your results too complex to visualise in isolation?

These six principles are relevant regardless of which visualisation you choose to create. In the following sections you will find guidance that is more specific to tables, graphs, and maps. While the guidance is specific to the different forms, they all tie into the central principles described here in this first section.

Chapter 3

Considering the Message

Two of the above principles go beyond the specifics of what you put into your data visualisations: effectiveness and being informative. Consideration of these two principles does not start when you plot your variables. They are principles which should guide your entire research process, including presentation.

This guide will make regular reference to considering what is appropriate for your message, your results and your purpose. For your data visualisations to be effective and informative, you need to think hard about the message you want them to convey. This will often come back to an original research question. These research questions should always be guiding you in the creation of data visualisations. Effective data presentation needs to have something to say, and what it says should be relevant.

Consider this as a process:

1. We start with our research questions that we want to help answer through our research.
2. We can break these up and consider how we will answer them. What are going to be the key points we will need to investigate to answer these questions?

For example, say we want to research the prevalence of a disease across areas within a country. We can decide that we are going to need to make points about the overall prevalence, the geographical variation, the explanations, compounding variables. We could look at these as the building blocks of our messages. Our messages are what we want people to remember and they will all stack up to help answer our bigger questions.

3. After this, we conduct our analysis and pick out our key findings. These key findings will similarly be informed by our existing research questions

and pre-conceived ideas about what our messages will be. However, they should always be flexible; an unexpected result should not be ignored.

4. We now need to update our messages based on what we have observed. Our messages should always strive to be important, relevant, and interesting. Also consider novelty; repeating a message we have heard many times over and over will not result in a very interesting data visualisation.

5. These updated messages and key findings will inform the creation of our presentable data visualisations. These visualisations along with our messages help to answer our initial research questions.

Therefore, we think about being effective and informative throughout the research process. If your messages and questions are not effective or informative then you cannot expect your visualisations to be.

Bringing effectiveness and informativeness into your data visualisations requires careful consideration of the messages you have drafted. This leads to questions you will need to ask of yourself, including:

- What variables should I include?

You should not be including more variables than are necessary. Think about the specifics of your intended message and only include the variables which are relevant and necessary for effectively showing this message. You should also avoid including variables which are uninformative. If adding in a variable does not add any explanatory value, then drop it from your visualisation.

- Which variables should I split by?

Disaggregating your findings by certain groups is a common practice. What variables you use to do this splitting should largely be informed by your research questions and messages.

Consider the example of geographical variation of disease prevalence. Explicitly, we know we will need to look at how our results vary by geography, but we may also want to consider variables which could help to explain the geographical variation. So, we could split by levels of economic deprivation or rurality. It always comes back to keeping relevant and important.

- Which graph/table/map should I use?

This involves thinking carefully about the type of message you want to show. Does your message mostly concern changes over time? If so, then a table probably is not suitable unless it is quite a short time frame with few points.

A few graphical options would be suitable including line graphs, column charts, slope charts etc. From here, the choice would now be dependent on the types of variables you want to show. Concerned with totals? Then consider column charts. Concerned with averages/rates? Consider a line chart.

Deciding on the right type of graph is a process, starting from your overall message and working down:

1. What is the purpose of your message? (Change over time? Distribution? Spatial? Correlation? Etc.)
2. What variables are you plotting and how many? (Categorical? Continuous?)
3. How much data needs displaying? A lot or a little?
4. What are the measurement units? (Averages? Totals? Rates? Proportions?)

Considering these questions should help you to narrow down what type of data visualisation is most appropriate. There are too many possible variations to consider here, but these points should help guide your thinking process.

Chapter 4

Tables

When thinking of data visualisations, tables may not be your first choice as they are not as visually remarkable as a graph. However, tables are a crucial tool in presenting data and results as they have the advantage of much greater specificity than graphs and are usually simple to understand. Generally, it is harder to read patterns in tables than in graphs. Therefore, graphs should be used when you want to focus on patterns, trends and relationships that do not necessarily require the exact values to be understood.

A table would therefore be more appropriate than a graph or map if:

- You are asking the audience to compare individual values directly
- You are wanting to include both the values and some derived measures such as percentages or indices. These are harder to show succinctly all together on one graph.
- You want to include summary statistics such as means or totals
- You need to show values with very different magnitudes together.
- If users may want to use the data for their own analysis or reference.

Reference tables contain extensive information for people to look up.

- They are useful for archival purposes rather than analysis.
- They should include detailed metadata about the information presented: what, where and when of the data.
- They usually appear as appendices.

Demonstration tables are probably what you think of when we mention tables for research purposes.

- They are intended to reinforce a point by showing statistics or values that can be quickly assimilated by the reader.
- They are included within the text to allow readers to follow the general argument and without having to flip back and forth to refer to the relevant information.
- It is important they are clear and well-presented, usually using reasonable approximations to keep figures to a few significant figures.
- Very large demonstration tables can be confusing and intimidating. If all the information is truly required, it should be split across multiple smaller tables.

The following guidance mostly concerns the formatting of demonstration tables although the general principles are applicable to both forms.

Reference tables however are not designed to draw attention to specific numbers, patterns, or comparisons and therefore advice on topics such as ordering of columns and rows are not especially relevant.

4.1 Introduction to Flextable

This section of the guide will be supplemented by reproducible code examples from R. These will be focusing on the incredibly useful package flextable which has been designed to help create report ready tables directly in R. It is especially useful for those intending to write their reports or knit their documents into Word format. This guide will mostly cover some of the basic features, for extended guidance on the full capabilities of flextable, please see this guide

In general, when using flextable the idea is to use R code to manipulate your data into roughly the format you wish to present as a table. In other words you create your table as a data frame. For purposes of demonstration, all data manipulation shown shall be done using the tidyverse range of packages, particularly dplyr and tidyr.

Once you have your data in a desired format you can apply flextable functions to the data to create and design your table. Starting with `flextable()` to turn your data from a data.frame/tibble object into a flextable object.

A flextable object consists of 3 parts.

- header: the section containing any and all headers/titles (defaults to column names of data frames in a single row)
- body: this contains all of the data from the data frame
- footer: not present by default but can be used to add footnotes or additional content

Additionally, in flextable notation `i` donates rows and `j` donates columns. Both can be referred to in functions by their row or column number. Note however that the numbering works within parts e.g. by default the row of column headers is i=1 within the part "header", while the first row of data is i=1 within the part "body".

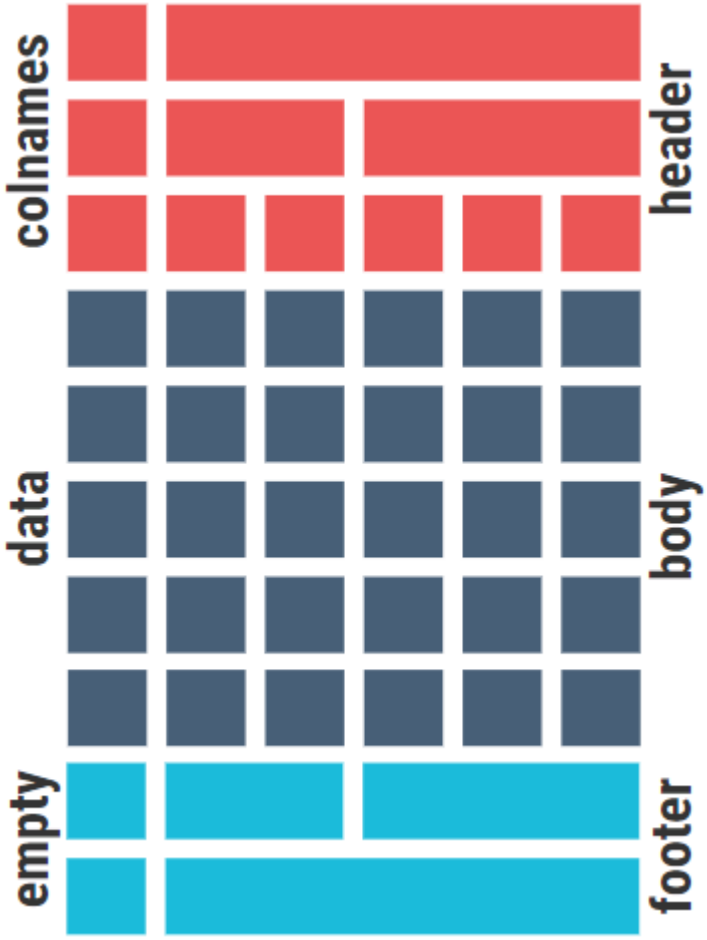

Figure 4.1: Structure of a flextable object

Flextable can be installed to R in the usual manner from CRAN.

```
#install.packages("flextable")

library(flextable)
```

4.2 General Guidance

4.2.1 Title, column headers and labels

Titles and labels are very important to the design of a table as they help users understand what is being presented. The titles and labels make sure the table works on its own and can be read within a different context than its original presentation.

You should consider including the following information in tables within either titles, labels, headings or possible footnotes, the choice of which points depends on your data and how important the details are to understanding the information:

- Analysis units (people, households etc.)
- Types of statistics (totals, means etc.)
- Units (thousands, kg, $)
- Geographical coverage
- Time period
- Source of data
- Key quality information

4.2.1.1 Using Flextable - Column headers and titles

Let's look at how we can start using the flextable package to create report ready tables. Starting with how we can set and edit the titles and column headers of our tables.

First we are going to read in our data.

The first dataset data we are using in this guide is an extract of a survey conducted in Uganda from farmers identified as growing beans.

The dataset contains an extract of 50 responses to 23 of the survey questions, and has been imported to R as a data frame called `BeanSurvey`.

A summary of the columns in the dataset is below.

```
BeanSurvey <- readRDS(file = "data/bean_survey.RDS")
```

We shall start by producing some basic summaries of the data (mean number of adults in the household and mean number of children in the household). We shall also split this by Village.

We can perform this data manipulation using the tidyverse range of packages. We shall save the summary data as `HHcomposition`.

Column	Description
ID	Farmer ID
VILLAGE	Village name
HHTYPE	Household composition
GENDERHH	Gender of Household Head
AGEHH	Age of Household Head
OCCUHH	Occupation of Household Head
ADULTS	Number of Adults within the household
CHILDREN	Number of Children (<18) within the household
MATOKE	Do they grow matoke?
MAIZE	Do they grow maize?
BEANS	Do they grow beans?
BANANA	Do they grow banana?
CASSAVA	Do they grow cassava?
COFFEE	Do they grow coffee?
LANDAREA	Land area of farm (acres)
LABOR	Labor usage
INTERCROP	Intercrops with beans
DECISIONS	Household decision responsibility
SELLBEANS	Do they grow beans for sale?
BEANSPLANTED_LR	Quantity of beans planted in long rain season
BEANSPLANTED_SR	Quantity of beans planted in short rain season
BEANSHARVESTED_LR	Quantity of beans harvested in long rain season
BEANSHARVESTED_SR	Quantity of beans harvested in short rain season

VILLAGE	ADULTS	CHILDREN
Kimbugu	2.1	2.2
Lwala	2.7	2.3

Village	Adults (mean)	Children (mean)
Kimbugu	2.1	2.2
Lwala	2.7	2.3

```
library(tidyverse)

HHcomposition <- BeanSurvey%>%
  group_by(VILLAGE)%>%
  summarise(ADULTS = mean(ADULTS, na.rm = TRUE),
            CHILDREN = mean(CHILDREN, na.rm = TRUE))
```

Now that the data is in a structure (2 X 3) that could be used as a presentable table, we can make the first step of turning this data.frame into a flextable object using `flextable()`. Flextable supports the use of the pipe operator `%>%` similar to other tidyverse packages.

```
HHcomposition%>%
  flextable()
```

The default column headers are of course the column names of the data but we can use additional functions to make these more presentable. First we can change the headers using `set_header_labels()`.

Note that our column names better explain the meanings of the columns and have now included the unit of measurements (mean) so that the numbers can be understood

```
HHcomposition%>%
  flextable()%>%
  set_header_labels(VILLAGE = "Village",
                    ADULTS = "Adults (mean)",
                    CHILDREN = "Children (mean)")
```

Next we can add a title using the function `add_header_lines()`.

Additionally we have used the functions `align()` to centre align our title and `bold()` to make it bold face. Note that in both functions `i = 1, part = "header"` is telling R to apply these functions to row of the header "part" of our table. The row of column headers is now row 2 within the header.

We can also use `autofit()` to automatically fit the table to a more appropriate width. If you are intending to use the tables in word documents, it would be

Household Composition Across Villages

Village	Adults (mean)	Children (mean)
Kimbugu	2.1	2.2
Lwala	2.7	2.3

advised to instead use `width()` to fit individual column widths as `autofit()` will not account for the width of the page.

```
HHcomposition%>%
  flextable()%>%
  set_header_labels(VILLAGE = "Village",
                    ADULTS = "Adults (mean)",
                    CHILDREN = "Children (mean)")%>%
  add_header_lines(values = "Household Composition Across Villages")%>%
  align(i = 1, align = "center", part = "header")%>%
  bold(i = 1, part = "header")%>%
  autofit()
```

4.2.2 Comparing numbers (rounding, decimal places and alignment)

Tables will require the reader to compare numbers. If these numbers are differently rounded or contain differing levels of significant figures/decimal places, then comparing them becomes more difficult. Here are some things you can do to make this process easier:

- The same level of precision should be used within each variable. The precision can vary between them, because different measures or ranges will require different levels of precision to make an accurate comparison, but it should be consistent within each variable.
- It is best to minimise the number of decimal places such that comparisons can be effectively made without any loss of information.
  - It is uncommon to need more than 3 decimal places. If you are dealing with incredibly small precise values, then consider using scientific notation. However, bear in mind that scientific notation is harder to understand for most readers.

- Rounding larger numbers is also advisable depending on your purpose. Demonstration tables usually use suitably rounded numbers that effectively illustrate the message. Reference tables tend to use a higher level of precision as users typically require a more exact number.
- Using commas to separate large numbers can make these numbers more easily readable, although if you are presenting large numbers you should also consider standardising the numbers into thousands, millions etc.
- Generally, numbers should also be right aligned, as should the column headings. The decimal point should line up.
- Decimal numbers between 0 and 1 (or 0 and -1 if negative) should start with a 0 and not a decimal point.

4.2.2.1 Using Flextable - Formatting Columns

There are numerous ways you could format columns using data manipulation techniques prior to creating a flextable object, including by rounding numeric variables using `round()` or `signif()`.

Alternatively, flextable comes with a range of column formatting functions;

- `colformat_num` for formatting numeric columns
- `colformat_int` for formatting integers
- `colformat_double` for formatting decimals
- `colformat_char` for formatting character columns
- `colformat_date` for formatting dates
- `colformat_datetime` for formatting date-times
- `colformat_lgl` for formatting boolean/logical variables

The numeric based functions above each contain a mix of the same arguments. As mentioned previously they also use i and j to determine the column and row numbers to apply the function to. In addition to these they allow the following arguments;

- `big.mark` - to format the separator in large numbers
- `decimal.mark` - to format the separator in decimal numbers (not available with `colformat_int`)
- `digits` - to format the number of decimal places (not available with `colformat_int`)

4.2.3 Orientation

A table's orientation can significantly affect its readability. It is much easier to compare numbers within a column than within a row. Therefore, if we intend

to compare numbers across groups according to several variables, the variables should define the columns and the groups should define the rows.

This is true of both demonstration and reference tables.

4.2.4 Order of rows and columns

Another way to improve the layout is to consider the ordering of rows and columns. If there is some logical ordering to the groups, maybe because it is an ordinal variable, then you should keep them in this order. However, if there is no logical order, it is advised to order them according to the most important variable. The most important variable will depend on your data and objectives.

Additionally, in cases when one of your groups is "none" or "other", it is often sensible to put these as the bottom rows. A "none" group often serves as a useful baseline to compare all other groups against. An "other" group is usually a combination of rare instances and lacks specified information so is rarely useful to a table's overall message.

Ordering of rows and columns is generally not so important when creating reference tables, as these are less likely be used for comparisons or to spot patterns. However, keeping the rows in some form of logical order will likely still help with the table's readability.

4.2.5 Borders

Borders should be using sparingly and only when necessary. They can be used to help separate parts of a table or groups of rows. However, using them too much just makes the table look cluttered and can interrupt numerical comparisons.

Therefore, borders should be avoided within the main body of the table and there should be no vertical lines. Horizontal lines should only be used to separate out a table's header and footer from the main table body and the page itself. Horizontal borders are effective between column headers if there is a hierarchical grouping between them.

4.2.6 Font

Be consistent with your font and ensure it is professional. It is recommended to use sans serif fonts such as Open Sans, Arial, Helvetica, Tahoma, or Verdana. Bold should only be used for titles and headings. Keep changes in font size to a minimum and avoid small fonts.

4.2.7 Grouping of rows and columns

Grouping is often useful to maximise the amount of information displayed while maintaining the table's effectiveness and readability. For example, you may use levels of more than one categorical variable to define your rows or you may wish to present more than one measure for a variable (such as wishing to show the mean, the sample size, and the standard deviation).

While there may be a temptation to put horizontal borders between the different groups of rows and vertical borders between the groupings of columns, this should be avoided. Instead, using white space between the groupings is a much neater alternative that effectively separates out the information and keeps focus within the groups.

4.2.8 Summary rows and columns

Summary rows and columns are quite useful for providing extra information that may be useful for interpretation. These should be placed at the bottom or right of the table unless they are the primary message of your table, in which case putting them first and then disaggregating is acceptable.

4.2.9 Other

Some other general tips include:

- You can use footnotes to provide additional contextual information including:
- Source
- Units of measurement
- Statistical information (such as level of significance)
- Any mitigating information that helps with interpreting the figures.
- If the table spans multiple pages, include the table's heading at the top of each page.
- Do not put the table in the middle of text. Ensure an effective and neat layout between your table and your text.
- If your table would only need two or fewer columns and a handful of rows, consider just writing this information out in text.

Chapter 5

Graphs

Chapter 6

Maps

Chapter 7

Accessibility

The following advice is adapted from advice on accessibility published by the Government Statistical service (GSS)

While we may have specific target audiences in mind when we write up our results and produce data visualisations, we should always aim to ensure inclusivity by making our content accessible to those with certain impairments. The GSS specifies impairments to vision, hearing, mobility and thinking & understanding as key areas to consider. For data visualisations, this primarily concerns those with vision and thinking impairments. These tips are designed to make your results as clear and readable as possible in general.

7.1 Tables

- Use column headers which explain the content of the columns, including measurement units where applicable.
- Include derived variables (e.g., totals) at the end of columns or rows.
  - Try to use more rows than columns.
- Write out or clearly explain any acronyms.
- Use table footnotes/captions to provide extra important information that cannot fit in the main body of the table.
- If you do not need to use exact numbers, consider rounding larger numbers.

7.2 Graphs/Maps

- Write out or clearly explain any acronyms.
- Make sure there is a clear distinction between lines on a line graph.

- Do not use red and green together as it is difficult for colour-blind people to distinguish between them.

7.3 Colour

Colour is one of the most useful tools for supplying extra information to maps and graphs, and sometimes even tables. It can be used to clearly highlight patterns and relationships that could be missed by a monochromatic visualisation. Adding colour can make visualisations more effective, but this only works if viewers can tell which colour is which. For instance, the graph presented here uses colours which are far too similar.

```
ggplot(MathsGrades, aes(x = MotherEduc, fill = Ever_fail))+
  geom_bar(position = "fill", width = 0.8)+
  scale_fill_manual(values = c("#02A6AC", "#1385AC"))+
  scale_y_continuous(labels = percent)+ # requires the package "scales"
  labs(x = "Mother's Education Level",
       y = "Count",
       title = "Figure 3: Prevelance of Exam Failures amongst\nStudents by Level of Mo
       fill = "Previous \nExam Failures?",
       caption = "Source: Example Data Source (2020)")+
  theme_minimal()
```

Previous advice already presented on the use of colour in graphs and maps (using it to highlight, use it sparingly etc.) are still relevant and contribute to ensuring accessibility. The advice presented here tends to focus on the choice of colours themselves.

- Ensure the colours are accessible.
  - Colour blindness affects an individual's ability to distinguish between certain colours. It affects men more commonly than women.
  - Most commonly, it affects the ability to distinguish reds and greens. Avoid using these colours together. Less commonly, but to be considered, is blue-yellow colour blindness. The two graphs below demonstrate how a colour-blind person may see a graph using red and green. The colours are much harder to distinguish.
  - Red-green colour palettes may also not be clear when printed in greyscale.
  - Blue palettes are a safe default starting point as they are colour blind safe and are visible in grayscale, as shown in the picture below.

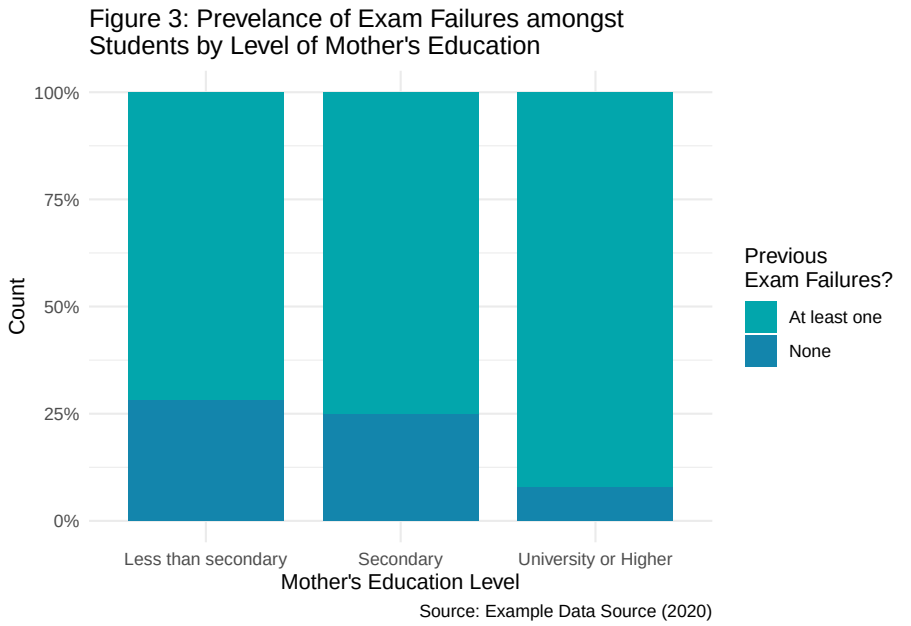

Figure 7.1: The colours set on this plot are far too similair

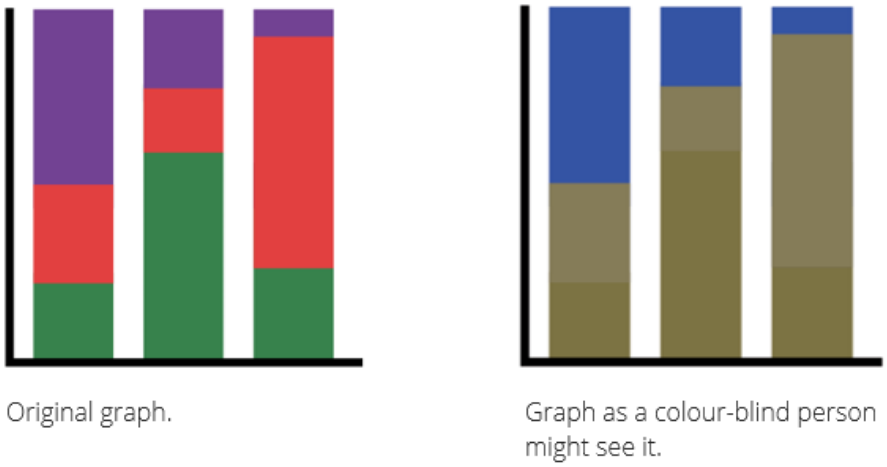

Figure 7.2: A demonstration of how a colour-blind person would see a colourful graph

Figure 7.3: Side by side comparison of colours converted to greyscale

- Choose colours carefully.
  - Consider cultural context. Colours often have some inherent culturally defined associations. For instance, using colours people associate with familiar concepts can improve the quality and speed of information processing, such as using blue for water on a map.

Figure 7.4: Using colours logically

- Understand the digital colour palette
  - Colours are represented using several common schemes. The most useful of these considers hue, saturation, and luminance. This scheme allows us to intuitively define unique colours.
- **Hue** – Hues are colours. They do not have a natural order and therefore users cannot assign a logical order to them. Small changes in hue are easy to detect although colour blindness can affect how well people can detect these differences.

Figure 7.5: Hue

- **Saturation** – This is the intensity of a colour, ranging from grey/white (no saturation) to rich, vibrant, almost glowing colour. Saturation is perceived on a continuous scale, although it is difficult to detect small changes. High saturation can also cause issues for those with certain visual/light sensitivity problems.

Figure 7.6: Saturation

- **Luminance** – the brightness of colour. Also perceived as continuous ordered scale from dark to light. This natural order can help us optimize colour schemes for maximum distinction and differentiation.
  - Changes are easier to detect
  - It is easier to distinguish between bars even if luminance is the only difference.
  - Changes in luminance need to be larger if creating a line graph as the white space between the lines makes it harder to quickly compare.

Figure 7.7: Luminance

- Never use an image as a background. This looks messy and cluttered and can make it difficult to read the graph.

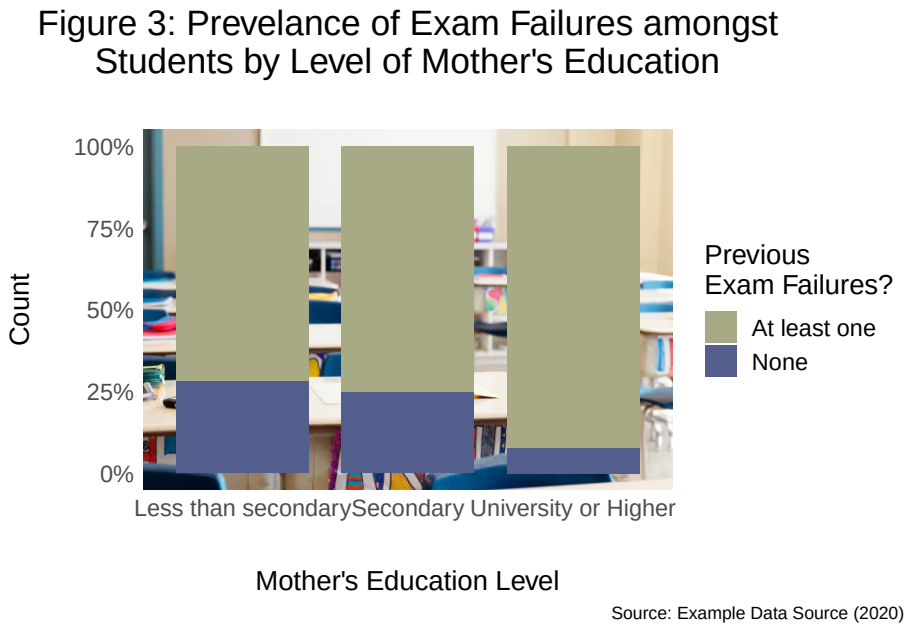

Figure 7.8: Using a background image in a plot

- Know how to use colour effectively
  - Alternate colours – consider alternating dark and light for categorical data to improve clarity and distinction.
  - Use borders – adding thin borders to the edges of bars can enhance clarity/separation.

- Avoid overuse of saturated colours.
  - Mid to low saturations are preferred
  - Only use bold saturated colour to draw attention to a specific piece of information or hard to see, small elements.
  - Bold, saturated colours can have visual side-effects. They may appear to glow for many users, can generate after-images and can affect how other colours appear.

```
ggplot(MathsGrades, aes(x = MotherEduc, fill = Ever_fail))+
  geom_bar(position = "fill", width = 0.8)+
  scale_fill_manual(values = c("#03F8FF", "#FF3360"))+
  scale_y_continuous(labels = percent)+
  labs(x = "Mother's Education Level",
```

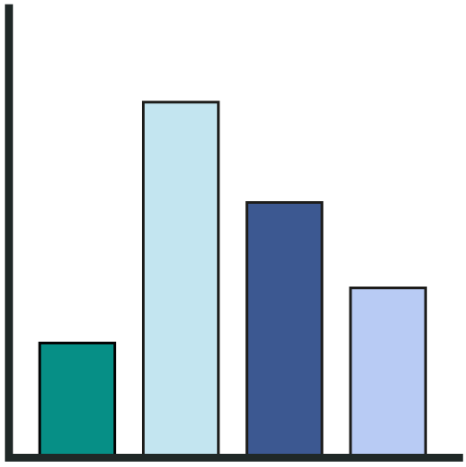

Figure 7.9: Use of alternating colours and borders

```
       y = "Count",
       title = "Figure 3: Prevelance of Exam Failures amongst\nStudents by Level of Mother's Educ
       fill = "Previous \nExam Failures?",
       caption = "Source: Example Data Source (2020)")+
  theme_minimal()
```

```
ggplot(MathsGrades, aes(x = MotherEduc, fill = Ever_fail))+
  geom_bar(position = "fill", width = 0.8, colour = "black")+
  scale_fill_manual(values = c("#60BDC0", "#C07384"))+
  scale_y_continuous(labels = percent)+
  labs(x = "Mother's Education Level",
       y = "Count",
       title = "Figure 3: Prevelance of Exam Failures amongst\nStudents by Level of Mother's Educ
       fill = "Previous \nExam Failures?",
       caption = "Source: Example Data Source (2020)")+
  theme_minimal()
```

- Use colour logically and consistently. If using sequences of colours, ensure that they progress in a logical manner that the user would expect, such as increasing luminance like in the map on page 50.
  - If creating multiple graphs, use the same colour to mean the same thing. Changing what they mean can confuse the user.

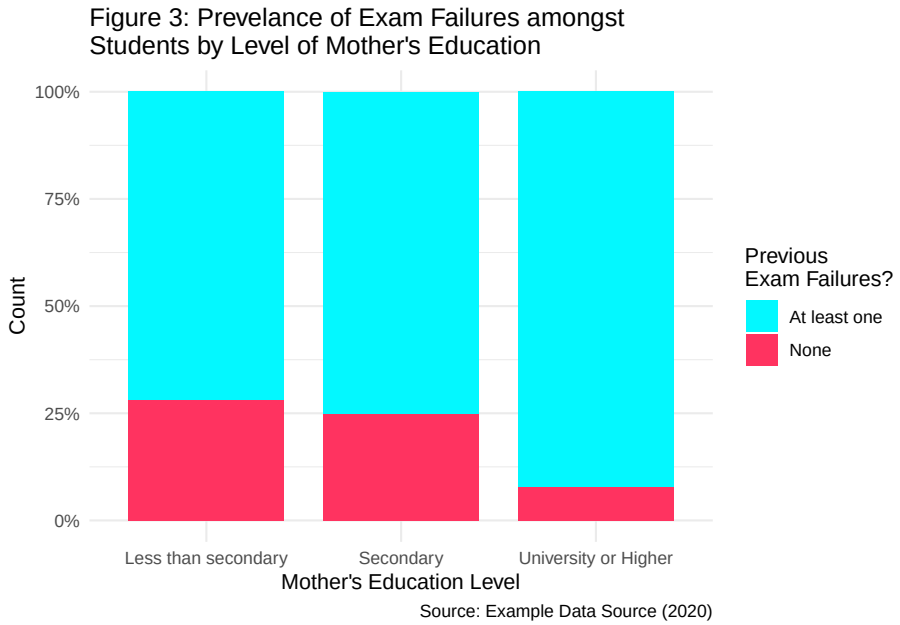

Figure 7.10: Over Saturated Colours on a graph

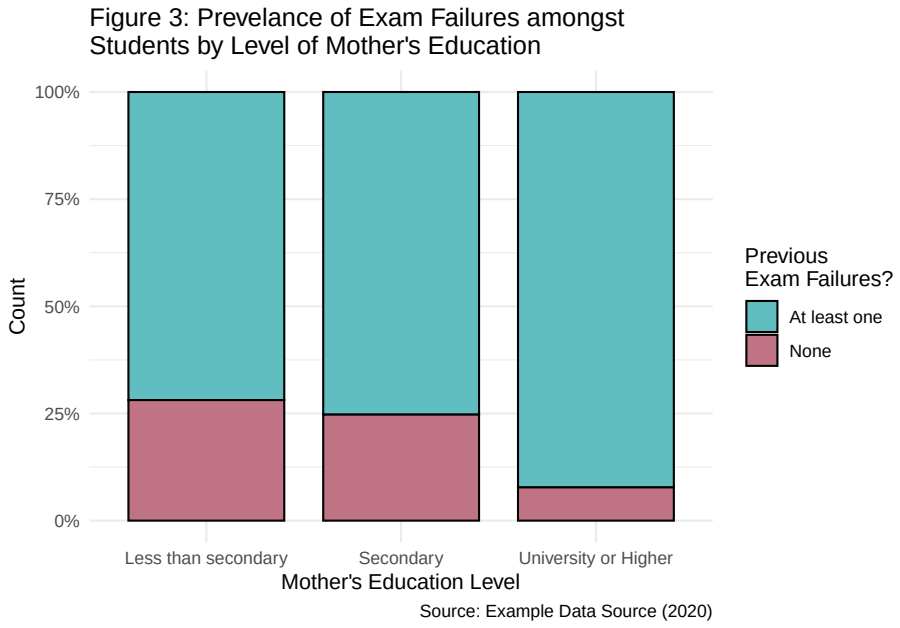

Figure 7.11: Low saturation makes the graph more pleasant to look at

- Use a white background.
  - Most palettes are designed to appear on top of a white background. It provides a helpful reference for the colour scale being used.

7.4 Using colour in ggplot2

ggplot2 comes with a wide array of capabilities to control the colour of various aspects of a graph. There are two main ways to add colour to a plot, either by allowing plotted objects (bar, points etc.) to vary according to a specified variable in your data or by simply setting something as a static colour.

To use a variable to define your colours, you include either `fill = variable` or `colour = variable` in the aesthetic mapping of your plot. `fill` refers to the space within objects/shapes such as the whitespace within a boxplot or a bar, while `colour` tends to refer to borders or points.

In the example below, we have assigned the `fill` of the column to vary according to the HHTYPE variable. Thereby turning the normal bar graph into a stacked bar graph.

```
ggplot(BeanSurvey, aes(x = VILLAGE, y = ADULTS, fill = HHTYPE))+
  geom_col()
```

As we can see below the same idea does not really work if we tried the colour argument instead.

```
ggplot(BeanSurvey, aes(x = VILLAGE, y = ADULTS, colour = HHTYPE))+
  geom_col()
```

Colour is better suited to graphs such as a scatterplot. In this case the plot has been jittered to avoid overplotting.

```
ggplot(MathsGrades, aes(x = Mock, y = Final, colour = Sex))+
  geom_jitter()
```

Again, this doesn't work when we try the reverse with fill.

```
ggplot(MathsGrades, aes(x = Mock, y = Final, fill = Sex))+
  geom_jitter()
```

If we don't wish to allow colour to vary according to a variable in our data but rather set all objects to the same colour (other than the default grey/black),

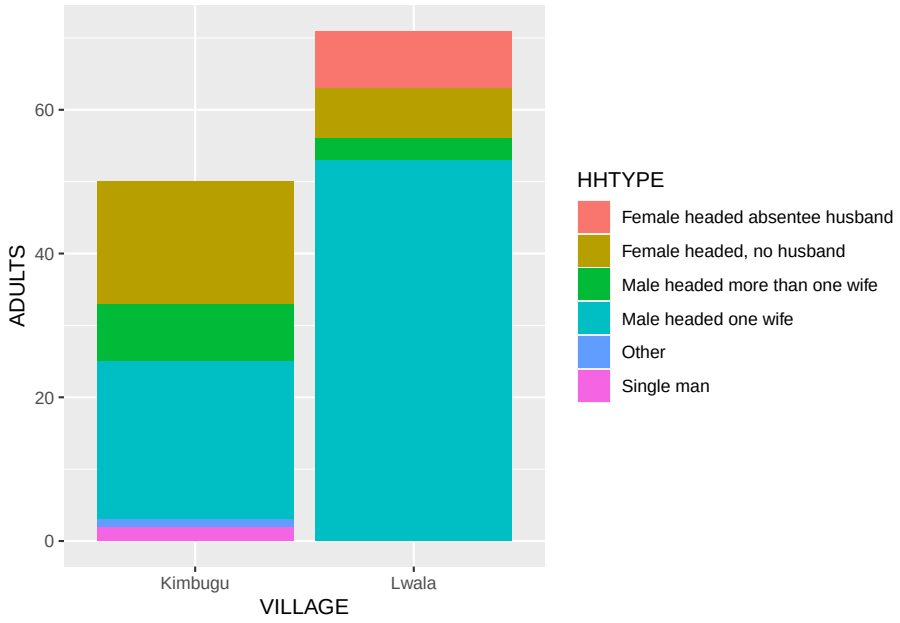

Figure 7.12: Using fill (barchart)

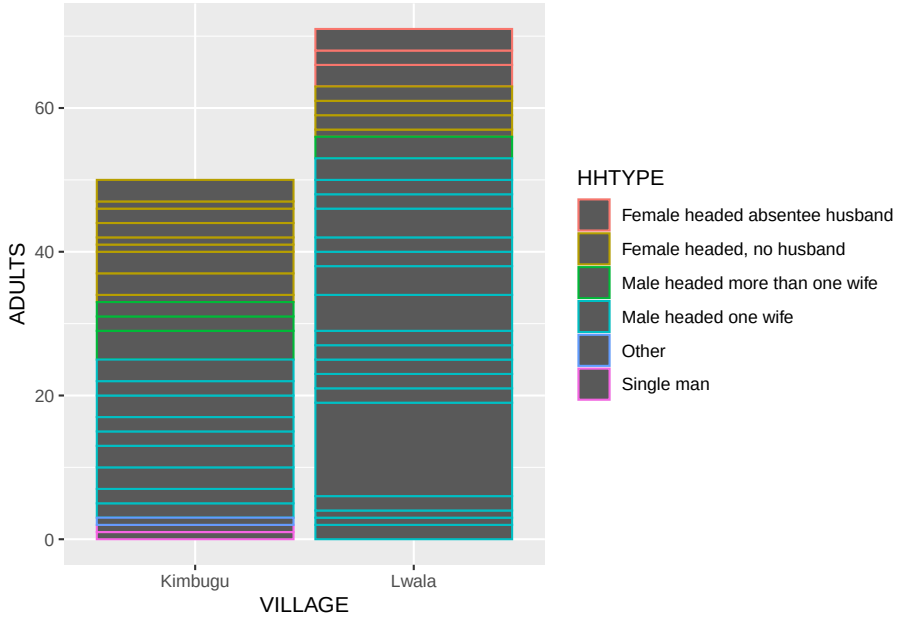

Figure 7.13: Using colour (barchart)

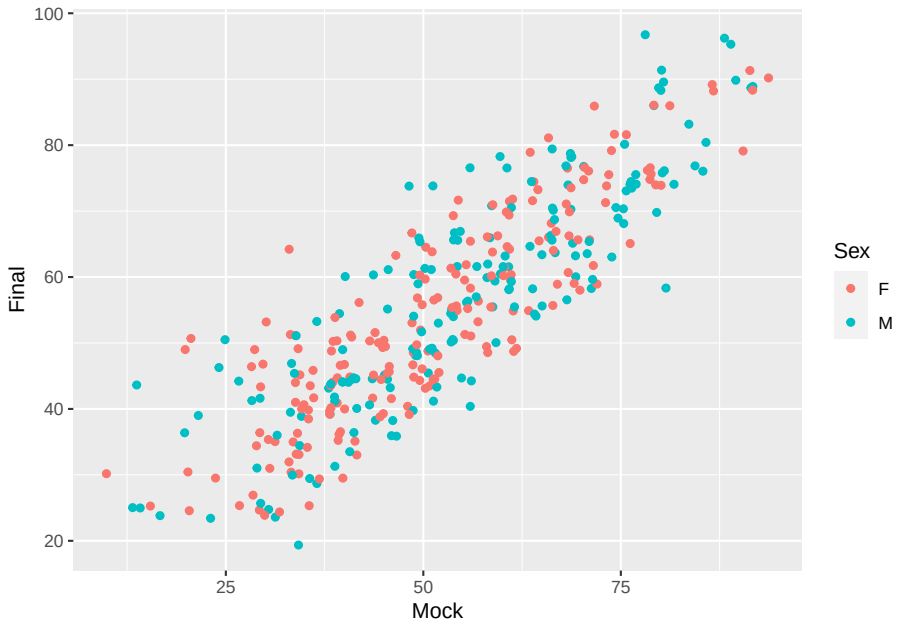

Figure 7.14: Using colour (scatter/jitterplot)

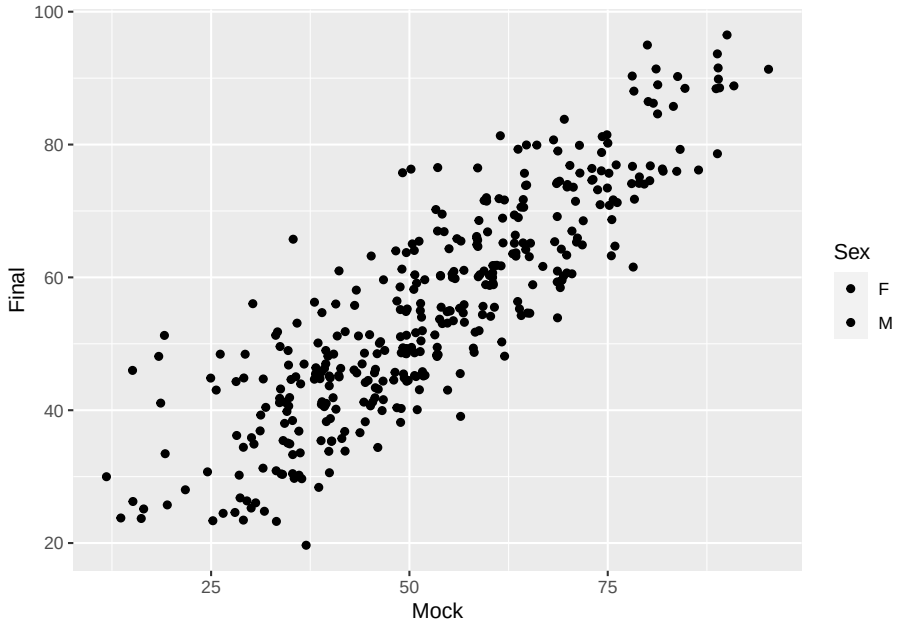

Figure 7.15: Using fill (scatter/jitterplot)

then we set the colour inside of the specific geom instead and not within a call to `aes()`. We can assign the colour using a hex code e.g. `fill = "#CD0000"` for a deep shade of red or by using the name of a pre-defined ggplot2 colour such as `colour = "dodgerblue4"`

```
ggplot(BeanSurvey, aes(x = VILLAGE, y = ADULTS))+
  geom_col(fill = "#CD0000")
```

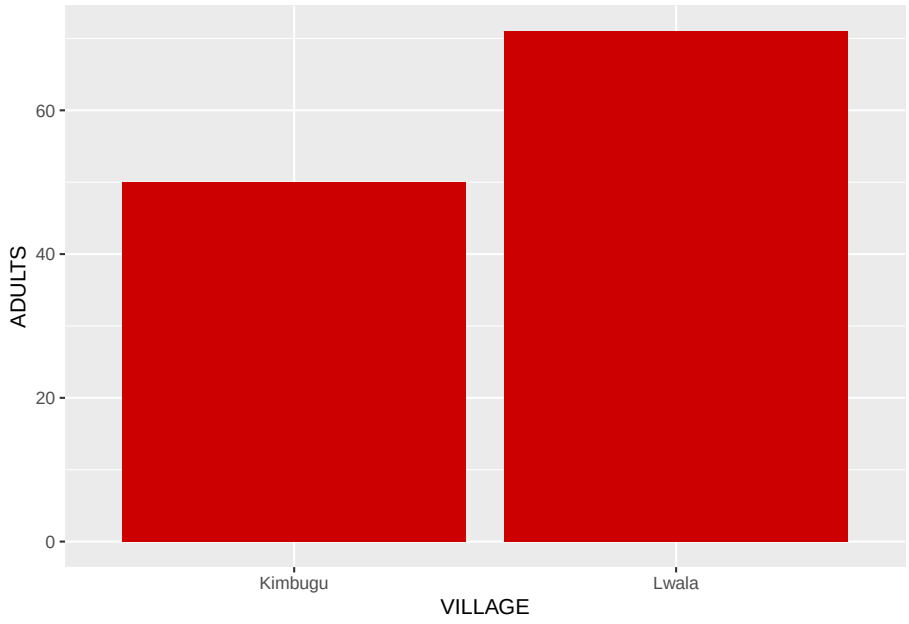

Figure 7.16: Setting fill with hex code

```
ggplot(MathsGrades, aes(x = Mock, y = Final))+
  geom_jitter(colour = "dodgerblue4")
```

7.4.1 Changing the colour/fill scale

7.4.2 Setting colours manually

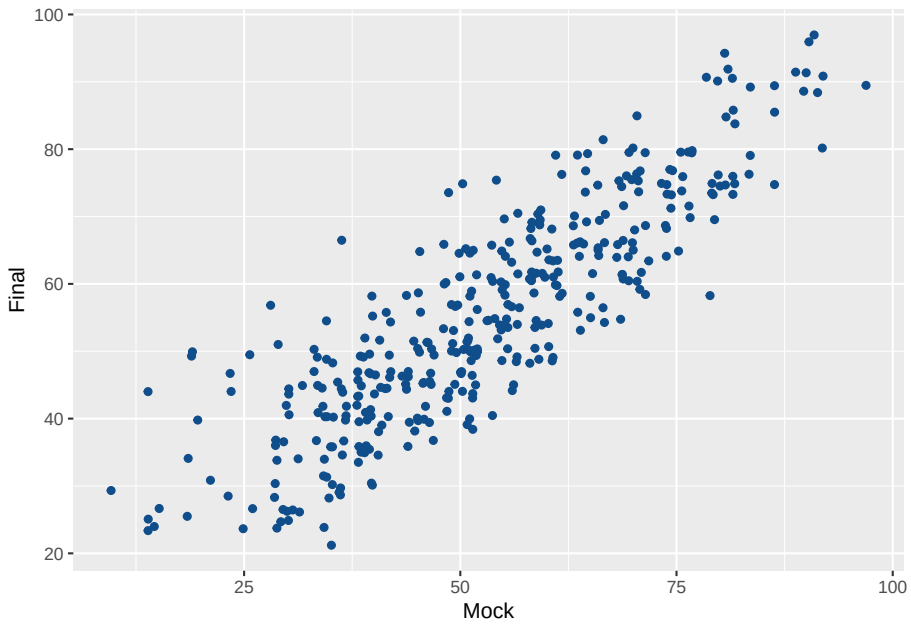

Figure 7.17: Setting colour with named colour