1. グラフを描く

> 何よりもまず，データを見せろ
>
> –Edward Tufte[*1]

データを可視化することは，データを分析しようとするものにとって最も重要な課題です。これが重要なのは，二つの異なる，しかし相互に関係し合う理由によります。まず，“提示するグラフ” を描くこととは，あなたのデータをスッキリと提示し，読者にとってあなたが言いたいことを簡単に理解させるために視覚的に訴えかけるようにすることです。同じぐらい，あるいはもっと重要なことは，グラフを描くことであなた自身がデータを理解できるようになることです。そのために，“探索的なグラフ” を描くことは，あなたがいざ分析しようとしているデータについて理解するのを助けることになるのが重要なのです。このことは当たり前のようでもありますが，私はこれを人に何回言ったかわからないほどです。

この章の重要さを示すために，優れたグラフというものがいかに有用なのかを示す典型例から始めたいと思います。そのために，図 **??**に最も有名なデータの可視化の例の一つを示しています。これは 1854 年，John Snow によるコロナの死亡者数の地図です。この図はその単純さにおいて，非常にエレガントだといえます。背景として，われわれは見る人の方向性を示すストリートマップを持っている，というのがあります。地図上には多数の小さな点があり，それぞれがコロナの発祥地点を表しています。大きな文字は水のポンプの位置を示していて，その名前ラベルがついています。この図をちょっと見ただけでも，アウトブレイクの源は Broad Street ポンプを中心にしていることが明らかです。このグラフを見て，Dr.Snow はポンプからハンドルを取り除き，500 人以上を殺したアウトブレイクを終わらせたのです。これが，良いデータの可視化の力です。

この章の目標は二つあります。まず，データを分析したり表示したりするとき，私たちがよく使うグラフについて説明し，続いてこれらのグラフを JASP で作成するにはどうすれば良いかを示します。このグラフそのものは，直接的なものなので，この章のある側面は非常にシンプルだと言えるでしょう。人がよく困惑するのは，グラフをどうやって作るかを学ぶとき，特に良いグラフをどうやっ

[*1] この言葉の原典は，Tufte の本『量的情報を可視化する』です。

て作れば良いかを学ぶときです。幸い，JASP でのグラフの書き方は，あなたがグラフの見え方にそれほどこだわらなければ，かなりシンプルなものです。私がこれをいうことの意味は，JASP のデフォルトのグラフがかなり良いものだということで，ほとんどの場合すっきりとクオリティの高いグラフィックを提供できるということです。しかし，標準的でない図を描きたいとあなたが思ったとき，あるいは図にかなり特殊な変更を加える必要があるとき，JASP のグラフィック関数は発展的な仕事や詳細な編集にはまだ向いていないということはあります。

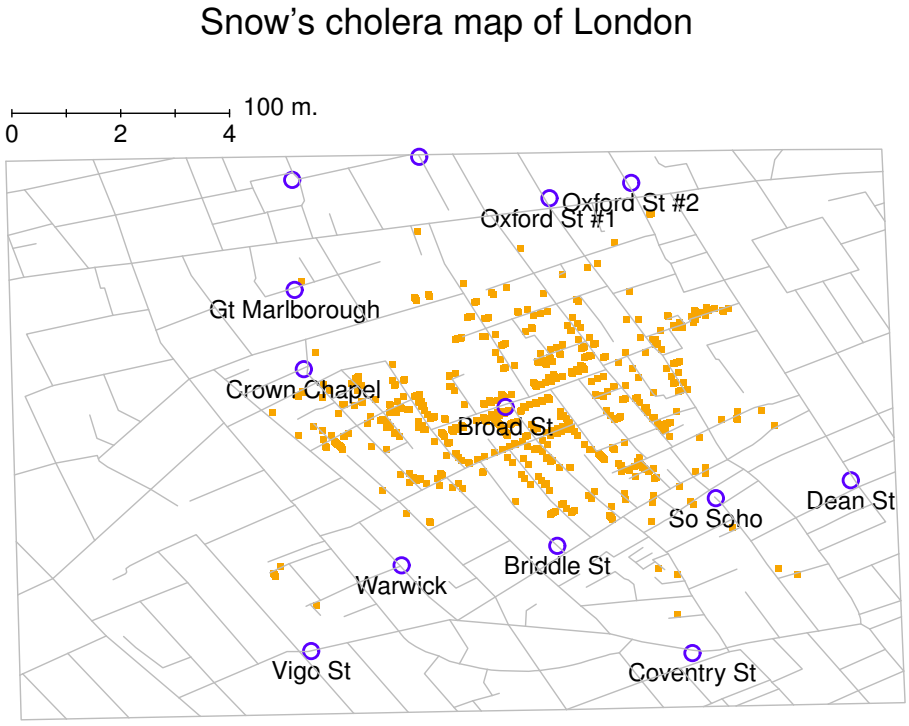

Figure1.1　John Snow のスタイリッシュなコロナマップのオリジナル。小さな各点はコロナ発生点で，大きな円は井戸の位置を示しています。このプロットが明らかにしたように，コロナのアウトブレイクは Broad St のポンプを中心にしていることがわかります。

1.1

Histograms

Let's begin with the humble **histogram**. Histograms are one of the simplest and most useful ways of visualising data. They make most sense when you have an interval or ratio scale variable (e.g., the `afl.margins` data from Chapter **??**) and what you want to do is get an overall impression of the variable. Most of you probably know how histograms work, since they're so widely used, but for the sake of completeness I'll describe them. All you do is divide up the possible values into **bins** and then count the number of observations that fall within each bin. This count is referred to as the frequency or density of the bin and is displayed as a vertical bar. Ihe AFL winning margins data there are 33 games in which the winning margin was less than 10 points and it is this fact that is represented by the height of the leftmost bar that we showed earlier in Chapter **??**, Figure **??**. The earlier graphs were generated using an advanced plotting package in R which, for now, is beyond the capability of JASP. But JASP gets us close, and drawing this histogram in JASP is pretty straightforward. Open up the 'Plots' menu under 'Descriptives' - 'Descriptive Statistics' and click the 'Distribution plots' check box, as in Figure **??**. JASP defaults to labelling the y-axis as 'Counts' and the x-axis with the variable name. The **bins** are selected automatically. Note that while counts are displayed, the actual values do not matter too much. Rather, what we are really interested in is our impression of the shape of the distribution: is it normally distributed or is there a skew or kurtosis? Our first impressions of these characteristics come from drawing a **histogram**.

One additional feature that JASP provides is the ability to plot a 'Density' curve on top of this histogram. You can do this by clicking the 'Display density' check box under the 'Plots' options, and this gives us the plot shown in Figure **??**. A density plot visualises the distribution of data over a continuous interval or time period. This chart is a variation of a histogram that uses **kernel smoothing** to plot values, allowing for smoother distributions by smoothing out the noise. The peaks of a density plot help display where values are concentrated over the interval. An advantage density plots have over histograms is that they are better at determining the distribution shape because they're not affected by the number of bins used (each bar used in a typical histogram). A histogram comprising of only 4 bins wouldn't produce a distinguishable enough shape of distribution as a 20-bin histogram would. However, with density plots, this isn't an issue.

Although this image would need a lot of cleaning up in order to make a good presentation graphic (i.e., one you'd include in a report), it nevertheless does a pretty good job of describing the data. In fact, the big strength of a histogram or density plot is that (properly used) it does show the entire spread of the data, so you can get a pretty good sense about what it looks like. The downside to histograms is that they aren't very compact. Unlike some of the other plots I'll talk about it's hard to cram 20-30 histograms into a single image without overwhelming the viewer. And of course, if

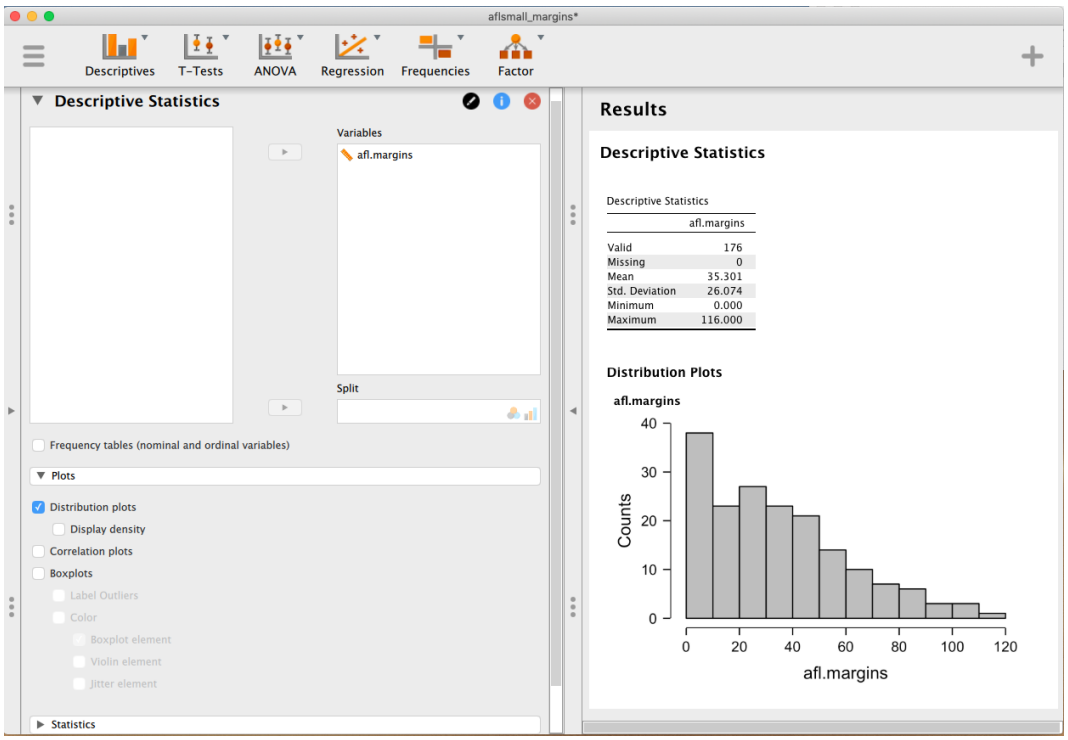

Figure1.2 JASP screenshot showing the 'Distribution plots' option and accompanying histogram.

your data are nominal scale then histograms are useless.

1.2 Boxplots

Another alternative to histograms is a **boxplot**, sometimes called a "box and whiskers" plot. Like histograms they're most suited to interval or ratio scale data. The idea behind a boxplot is to provide a simple visual depiction of the median, the interquartile range, and the range of the data. And because they do so in a fairly compact way boxplots have become a very popular statistical graphic, especially during the exploratory stage of data analysis when you're trying to understand the data yourself. Let's have a look at how they work, again using the `afl.margins` data as our example.

The easiest way to describe what a boxplot looks like is just to draw one. Click on the 'Boxplots' check box and you will get the plot shown on the lower right of Figure **??**. By default, JASP has

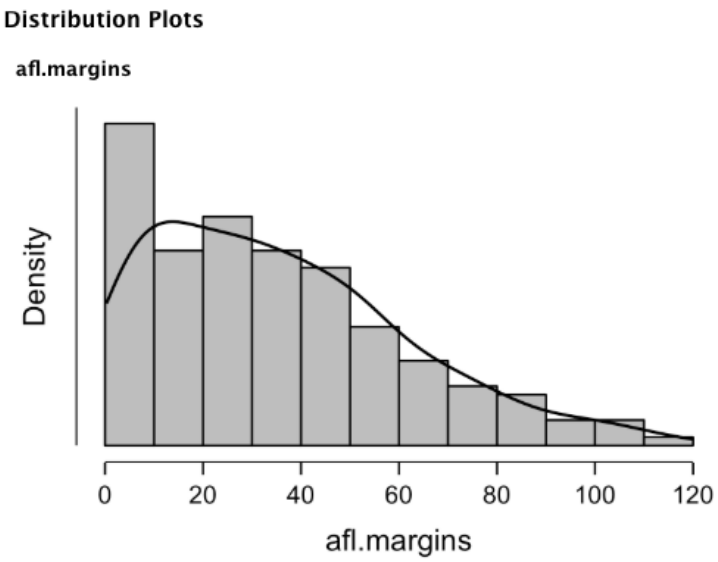

Figure1.3 A density plot of the `afl.margins` variable plotted in JASP

drawn the most basic boxplot possible. When you look at this plot this is how you should interpret it: the thick line in the middle of the box is the median; the box itself spans the range from the 25th percentile to the 75th percentile; and the "whiskers" go out to the most extreme data point that doesn't exceed a certain bound. By default, this value is 1.5 times the interquartile range (IQR), calculated as `25th percentile - (1.5*IQR)` for the lower boundary, and `75th percentile + (1.5*IQR)` for the upper boundary. Any observation whose value falls outside this range is plotted as a circle or dot instead of being covered by the whiskers, and is commonly referred to as an **outlier**. For our AFL margins data there are two observations that fall outside this range, and these observations are plotted as dots (the upper boundary is 107, and looking over the data column in the spreadsheet there are two observations with values higher than this, 108 and 116, so these are the dots).

1.2.1 Violin plots

A variation to the traditional box plot is the violin plot. Violin plots are similar to box plots except that they also show the kernel probability density of the data at different values. Typically, violin plots will include a marker for the median of the data and a box indicating the interquartile range, as in standard box plots. In JASP you can achieve this sort of functionality by checking

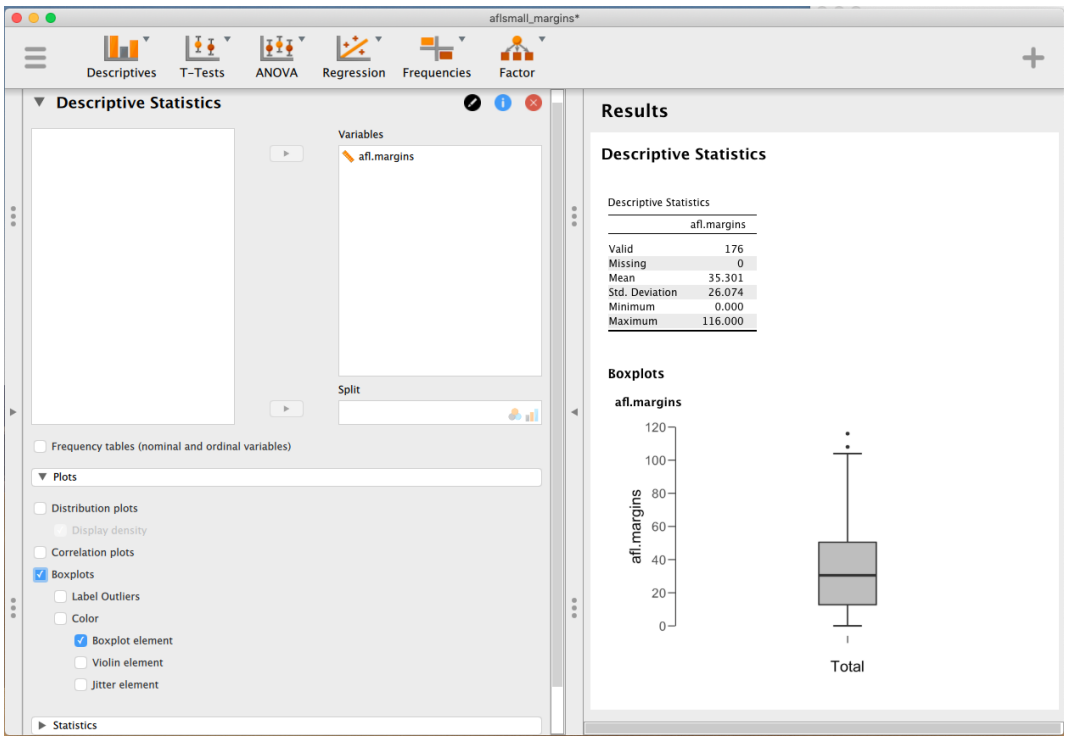

Figure1.4 A box plot of the `afl.margins` variable plotted in JASP

. . . . . .

both the 'Violin element' and the 'Boxplot element' check boxes. See Figure **??**, which also has the data plotted (select the 'Jitter element' check box to show the actual data points on the plot).

1.2.2 Drawing multiple boxplots

One last thing. What if you want to draw multiple boxplots at once? Suppose, for instance, I wanted separate boxplots showing the AFL margins not just for 2010 but for every year between 1987 and 2010. To do that the first thing we'll have to do is find the data. These are stored in the `aflsmall2.csv` file. So let's load it into JASP and see what is in it. You will see that it is a pretty big data set. It contains 4296 games and the variables that we're interested in. What we want to do is have JASP draw boxplots for the `margin` variable, but plotted separately for each `year`. The way to do this is to first change `year` to a nominal variable, then `year` across into the 'Split' box.

The result is shown in Figure **??**. This version of the box plot, split by year, gives a sense of why it's sometimes useful to choose box plots instead of histograms. It's possible to get a good sense of what the data look like from year to year without getting overwhelmed with too much

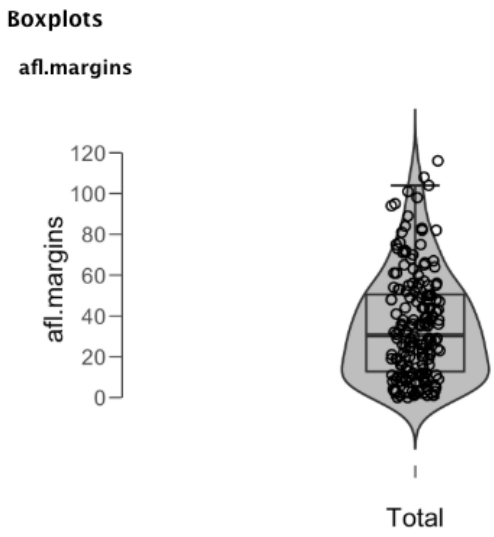

Figure1.5 A violin plot of the `afl.margins` variable plotted in JASP, also showing a box plot and data points

. . . . . .

detail. Now imagine what would have happened if I'd tried to cram 24 histograms into this space: no chance at all that the reader is going to learn anything useful.

1.3 Saving image files using JASP

Hold on, you might be thinking. What's the good of being able to draw pretty pictures in JASP if I can't save them and send them to friends to brag about how awesome my data is? How do I save the picture? Simples. Just click on the triangle beside the header of your plot and select 'Save Image As'. You can save it in one of several formats, including 'png', 'pdf', 'eps', or 'tif'. These formats all produce nice images that you can the send to your friends, or (perhaps more importantly) include in your assignments or papers.

1.4

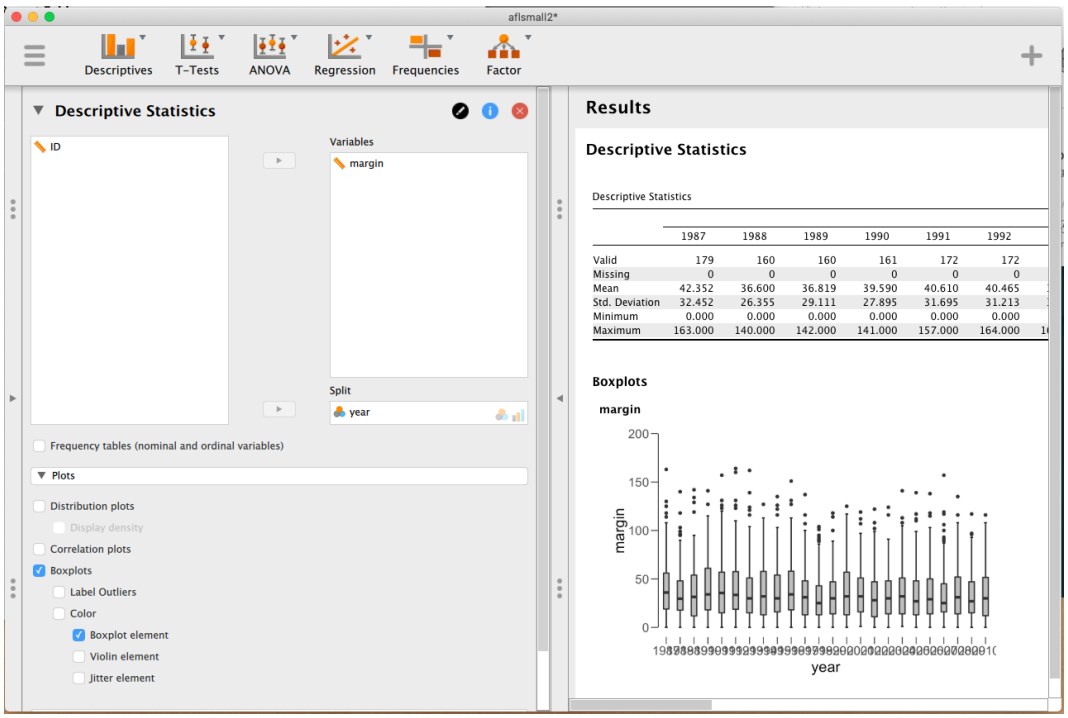

Figure1.6 Multiple boxplots plotted in JASP, for the `margin` by `year` variables in the `aflsmall2` data set

Summary

Perhaps I'm a simple minded person, but I love pictures. Every time I write a new scientific paper one of the first things I do is sit down and think about what the pictures will be. In my head an article is really just a sequence of pictures linked together by a story. All the rest of it is just window dressing. What I'm really trying to say here is that the human visual system is a very powerful data analysis tool. Give it the right kind of information and it will supply a human reader with a massive amount of knowledge very quickly. Not for nothing do we have the saying "a picture is worth a thousand words". With that in mind, I think that this is one of the most important chapters in the book. The topics covered were:

- *Common plots*. Much of the chapter was focused on some of the standard graphs that statisticians like to produce: histograms (Section **??**) and boxplots (Section **??**)
- *Saving image files*. Importantly, we also covered how to export your pictures (Section **??**)

One final thing to point out. While JASP produces some really neat default graphics, editing the

plots is currently not possible. For more advanced graphics and plotting capability the packages available in R are much more powerful. One of the most popular graphics systems is provided by the `ggplot2` package (see `http://ggplot2.org/`), which is loosely based on "The grammar of graphics" (**Wilkinson2006**). It's not for novices. You need to have a pretty good grasp of R before you can start using it, and even then it takes a while to really get the hang of it. But when you're ready it's worth taking the time to teach yourself, because it's a much more powerful and cleaner system.