Exploratory Data Analysis

CITS4009 Computational Data Analysis

Dr. Mubashar Hassan

Department of Computer Science and Software Engineering
The University of Western Australia

Semester 2, 2024

Section 1

Visualisation

What is visualisation?

Pictures are often better than text.

> *We cannot expect a small number of numerical values [summary statistics] to consistently convey the wealth of information that exists in data. Numerical reduction methods do not retain the information in the data. -William Cleveland*

The use of graphics to examine data is called **visualization**.

William Cleveland's Graphic Philosophy

- **A fine balancing act.**
  - A graphic should display as much information as it can, with the lowest possible cognitive strain to the viewer.
- **Strive for clarity.** Make the data stand out. Specific tips for increasing clarity include:
  - Avoid too many superimposed elements, such as too many curves in the same graphing space.
  - Find the right aspect ratio and scaling to properly bring out the details of the data.
  - Avoid having the data all skewed to one side or the other of your graph.
- **Visualization is an iterative process.** Its purpose is to answer questions about the data.
  - Different graphics are best suited for answering different questions.

Section 2

Exploratory Data Analysis (EDA)

What is exploratory data analysis?

Exploratory data analysis, or EDA for short, is a task that uses visualisation and transformation to explore your data in a systematic way.

EDA is an iterative cycle that involves:

- Generating questions about your data.
- Searching for answers by *visualising*, *transforming*, and *modelling* your data.
- Using what you learn to refine your questions and/or generate new questions.

EDA is not a formal process with a strict set of rules. More than anything, EDA is a state of mind.

The **goal** during EDA is to develop an understanding of your data.

The easiest way to achieve the goal is to use *questions* as tools to guide your investigation.

How to ask good questions?

EDA is fundamentally a creative process.

Like most creative processes, the key to asking *quality* questions is to generate a large *quantity* of questions.

but if there are too many low quality questions will not effect?

Two types of questions will always be useful for making discoveries within your data:

- What type of **variation** occurs within my variables?
- What type of **covariation** occurs between my variables?

Terms used in EDA

- A **variable** is a quantity, quality, or property that you can measure.
- A **value** is the state of a variable when you measure it. The value of a variable may change from measurement to measurement.
- An **observation** is a set of measurements made under similar conditions (you usually make all of the measurements in an observation at the same time and on the same object).
    - An observation will contain several values, each associated with a different variable.
    - An observation is also referred to as a data point.
- **Tabular data** is a set of values, each associated with a variable and an observation. Tabular data is tidy if each value is placed in its own "cell", each variable in its own column, and each observation in its own row.

What to look for in histograms and bar charts?

In both bar charts and histograms, tall bars show the common values of a variable, and shorter bars show less-common values.

Places that do not have bars reveal values that were not seen in your data.

To turn this information into useful questions, look for anything unexpected:

- Which values are the most common? Why?
- Which values are rare? Why? Does that match your expectations?
- Can you see any unusual patterns? What might explain them?

Does the data form subgroups?

Clusters of similar values suggest that subgroups exist in your data. To understand the subgroups, ask:

- How are the observations within each cluster similar to each other?
- How are the observations in separate clusters different from each other?
- How can you explain or describe the clusters?
- Why might the appearance of clusters be misleading?

Comparing two or more variables

- *Variation* describes the behavior within a variable,
- *Covariation* describes the behavior between variables.

Covariation is the tendency for the values of two or more variables to vary together in a related way.

The best way to spot covariation is to visualise the relationship between two or more variables. How you do that should again depend on the type of variables involved. If you have

- **a continuous variable and a categorical variable** – the categorical variable can be used as *legend*, *aesthetic mapping*;
- **two categorical variables** – try `geom_count` and `geom_tile`;
- **two continuous variables** – try `geom_point` and `geom_boxplot` and `geom_bin2d` or `geom_hex`.

Questions to ask for Covariation

Patterns in your data provide clues about relationships. If a systematic relationship exists between two variables it will appear as a pattern in the data. If you spot a pattern, ask yourself:

- Could this pattern be due to coincidence (i.e. random chance)?
- How can you describe the relationship implied by the pattern?
- How strong is the relationship implied by the pattern?
- What other variables might affect the relationship?
- Does the relationship change if you look at individual subgroups of the data?

References

- **Practical Data Science with R**, *Nina Zumel, John Mount*, Manning, 2nd Ed., 2020 (Chapter 3)
- **R for Data Science**, *Hadley Wickham, Garrett Grolemund*, O'Reilly, 2017 (Chapter 3)

Histogram and Density Plot

CITS4009 Computational Data Analysis

Dr. Mubashar Hassan

Department of Computer Science and Software Engineering
The University of Western Australia

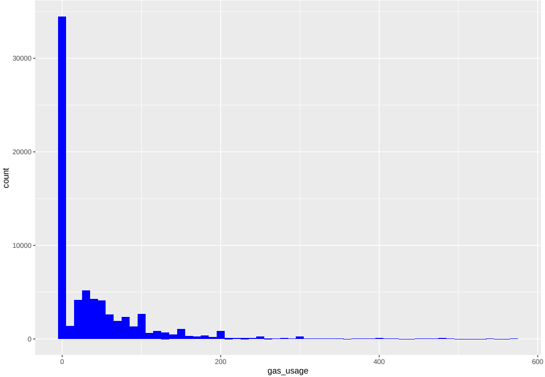
Semester 2, 2024

Section 1

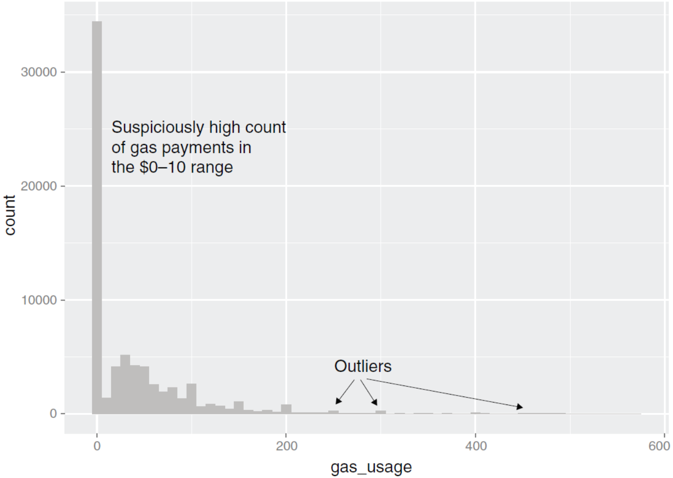
Histogram

Histogram

```
custdata_v2 <- readRDS('../../data_v2/Custdata/custdata.RDS')
library(ggplot2)
ggplot(custdata_v2, aes(x=gas_usage)) +
geom_histogram(binwidth=10, fill="blue")
```

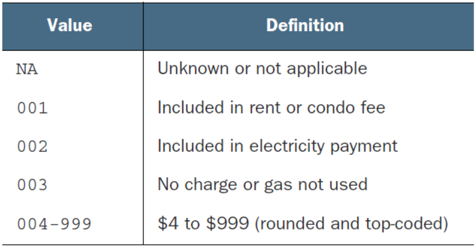


Reading a histogram



Most customers do not have gas heating?

Mixture of Numerical and Sentinel Values



Value	Definition
NA	Unknown or not applicable
001	Included in rent or condo fee
002	Included in electricity payment
003	No charge or gas not used
004-999	$4 to $999 (rounded and top-coded)

- The values in the `gas_usage` column are a mixture of numerical values and symbolic codes encoded as numbers.
- Options to deal with such cases:
  - Convert numerical values 1-3 to NA, and
  - Add additional Boolean variables to indicate the possible cases.

Histogram Take-Away

- With the proper `binwidth`, histograms visually highlight where the data is *concentrated*, and point out the presence of potential *outliers and anomalies*.
- The primary disadvantage of histograms is that you must decide ahead of time how wide the bins are:
  - Bins too wide - you can lose information about the shape of the distribution.
  - Bins too small - the histogram can look too noisy to read easily.

Section 2

Density Plot

Density Plot - *A Continuous Histogram*

Can think of a *density plot* as a *continuous histogram* of a variable, except the area under the density plot is rescaled to equal one.

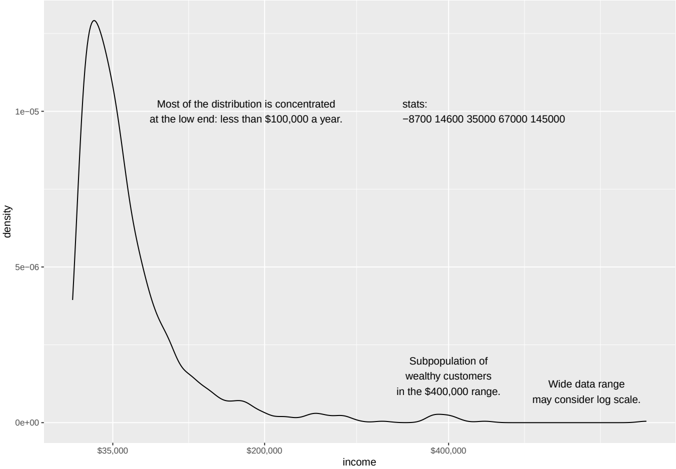
- A point on a density plot corresponds to the *fraction* of data (or the percentage of data, divided by 100) that takes on a particular value.
- This fraction is usually very small.
- When looking at a density plot, we should be more **interested in the overall shape** of the curve than the actual values on the y-axis.

Reading a plot and add annotation

```r
custdata_v1 <- read.table('../../data/custdata.tsv',header=T,sep='\t')
income_stat <- boxplot.stats(custdata_v1$income)$stats
income_stat_str <- paste(income_stat, collapse=" ")
library(scales)
fig <- ggplot(custdata_v1) + geom_density(aes(x=income)) +
  labs(y="density") +
  scale_x_continuous(labels=dollar, breaks=c(35000,200000,400000)) +
  annotate("text", x = 180000, y = 1e-05,
    label = paste("Most of the distribution is concentrated",
      "at the low end: less than $100,000 a year.", sep="\n")) +
  annotate("text", x = 400000, y = 1.5e-06,
    label = paste("Subpopulation of", "wealthy customers",
      "in the $400,000 range.", sep="\n")) +
  annotate("text", x = 550000, y = 1e-06,
    label = paste("Wide data range", "may consider log scale.",
     sep="\n")) +
  annotate("text", x=350000, y = 1e-05, hjust=0,
    label=paste("stats: ", income_stat_str, sep="\n"))
```

The annotated plot

```
fig
```



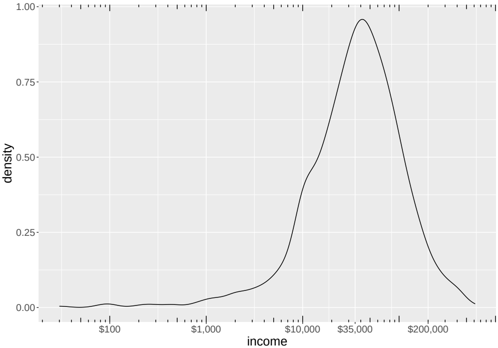
When should we use a logarithmic scale

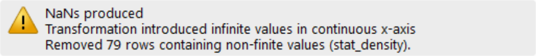
One should use a logarithmic scale when percent change or change in orders of magnitude is more important than changes in absolute units.

- In other words, the absolute unit changes can be interpreted differently in different context.
    - For example, in income data, a $5,000 difference in income means something very different in a population where the incomes tend to fall in the $10,000 range, than it does in populations where incomes fall in the $100,000-$1000,000 range.
    - What constitutes a "significant difference" depends on the order of magnitude of the incomes you're looking at.
- A log scale should be used to better visualize data that is heavily skewed.
    - For example, a few people with very high income will cause the majority of the data to be compressed into a relatively small area of the graph.

Plotting on a logarithmic scale

```
ggplot(custdata_v1) + geom_density(aes(x=income)) +
  scale_x_log10(breaks=c(100,1000,10000,35000,200000),labels=dollar) +
  annotation_logticks(sides="bt") + theme(text = element_text(size = 18))
```





NaNs produced
Transformation introduced infinite values in continuous x-axis
Removed 79 rows containing non-finite values (stat_density).

References

- **Practical Data Science with R**, *Nina Zumel, John Mount*, Manning, 2nd Ed., 2020 (Chapter 3)
- **R for Data Science**, *Hadley Wickham, Garrett Grolemund*, O'Reilly, 2017 (Chapter 3)
- Understanding and Interpreting Box Plots: https://www.wellbeingatschool.org.nz/information-sheet/understanding-and-interpreting-box-plots

Box Plot

CITS4009 Computational Data Analysis

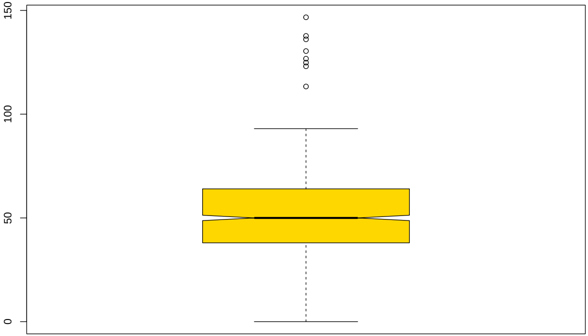
Dr. Mubashar Hassan

Department of Computer Science and Software Engineering
The University of Western Australia

Semester 2, 2024

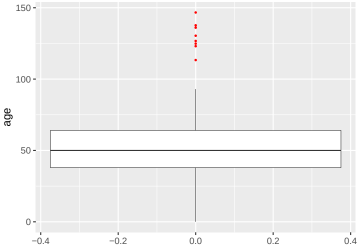
Boxplot (R)

```r
custdata<- read.table('../../data/custdata.tsv',
                                    header=T,sep='\t')
boxplot(custdata$age, notch=TRUE, col="gold")
```



Boxplot (ggplot)

```
ggplot(custdata) +
  geom_boxplot(aes(y=age), outlier.colour="red",
      outlier.shape=16, outlier.size=2, notch=FALSE)
```



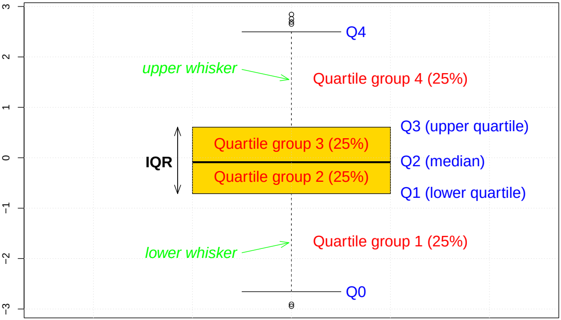
```
boxplot.stats(custdata$age)$stats
## [1]  0 38 50 64 93
```

Components of a box plot

- **Median**
  - The median (middle quartile) marks the mid-point of the data and is shown by the line that divides the box into two parts. Half data points are greater than or equal to this value and half are less.
- **Inter-quartile range**
  - The middle “box” represents the middle 50% of data points for the group. The range from lower to upper quartile is referred to as the *inter-quartile range* (IQR).
- **Upper quartile**
  - Seventy-five percent of the data points fall below the upper quartile.
- **Lower quartile**
  - Twenty-five percent of the data points fall below the lower quartile.
- **Whiskers**
  - The upper and lower whiskers represent data outside the middle 50%. Whiskers often (but not always) stretch over a wider range than the middle quartile groups.

Components of a box plot

```
x <- rnorm(1000); boxplot(x, col="gold"); grid()
cat(boxplot.stats(x)$stats) # we call these numbers Q0,...,Q4
```

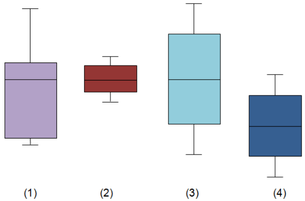


```
## -2.656523 -0.711676 -0.08816674 0.6061266 2.497597
```

Interpreting a box plot

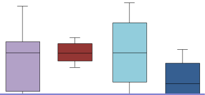
Imagine these are box plots of students' exam marks for different units.

- When a box plot is comparatively short:
  - See example (2), this suggests that overall the students' marks do not vary greatly.
- When a box plot is comparatively tall:
  - See examples (1) and (3). These two plots suggest that the students' marks do vary a lot.
- When one box plot is much higher or lower than another:
  - Example: Unit 3's marks are higher than unit 4's.



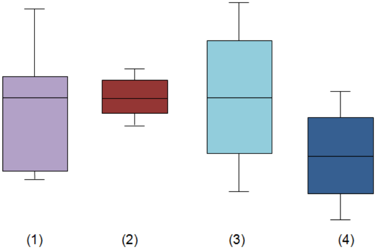
Interpreting a box plot

- Obvious differences between box plots:
  - See examples (1) and (2), (1) and (3), or (2) and (4).
  - Any obvious difference between box plots for comparative groups is worthy of further investigation in the *Items at a Glance* reports, e.g., the IQR for a unit's exam marks is much higher or lower than the IQR for the department's reference group box plot.
- When the four sections of the box plot are uneven in size:
  - See example (1). The median is skewed to one side of the box. This shows that many students have similar marks in quartile group 3 but their marks vary greatly in quartile group 2. The long upper whisker means that students' marks also vary a lot in quartile group 4; the very short lower whisker means that the marks are very similar in quartile group 1.



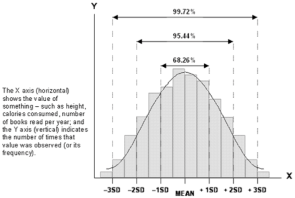
Same median, different distribution

- See examples (1), (2), and (3). The medians (which generally will be close to the average) are all at the same level.
- However the box plots in these examples show very different distributions of students' exam marks.

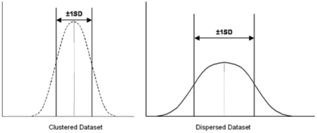


Mean and Std

- The mean is the most commonly used mathematical measure of average and is generally what is being referred to when people use the term “average” in everyday’s language.
  - The mean is calculated by totalling all the values in a dataset; this total is then divided by the number of values that make up the dataset.
- The standard deviation is a measure that summarises the amount by which every value within a dataset varies from the mean.
  - Effectively it indicates how tightly the values in the dataset are bunched around the mean value.



The X axis (horizontal) shows the value of something – such as height, calories consumed, number of books read per year; and the Y axis (vertical) indicates the number of times that value was observed (or its frequency).

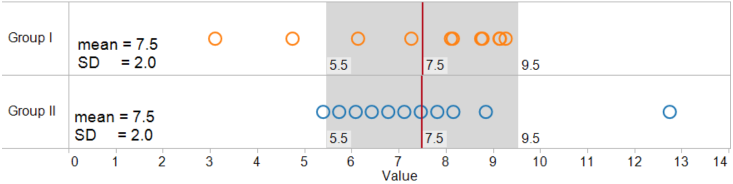


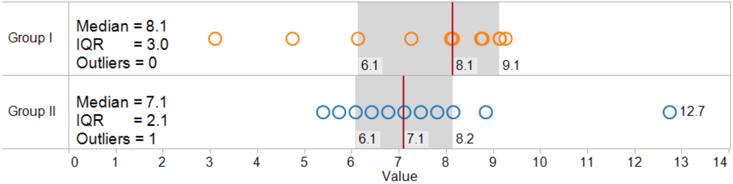
Why median and IQR are better than mean and standard deviation?

Like mean and standard deviation, median and IQR measure the central tendency and spread, respectively, but are robust against outliers and non-normal data. They have a couple of additional advantages:

- Outlier Identification. IQR makes it easy to do an initial estimate of outliers:
  - $< Q1 - 1.5 * IQR$
  - $> Q3 + 1.5 * IQR$
- Skewness. Comparing the median to the quartile values shows whether data is skewed.

Why median and IQR are better than mean and standard deviation?





References

- **Practical Data Science with R**, *Nina Zumel, John Mount*, Manning, 2nd Ed., 2020 (Chapter 3)
- **R for Data Science**, *Hadley Wickham, Garrett Grolemund*, O'Reilly, 2017 (Chapter 3)
- Understanding and Interpreting Box Plots: https://www.wellbeingatschool.org.nz/information-sheet/understanding-and-interpreting-box-plots

Bar Chart and Dot Plot

CITS4009 Computational Data Analysis

Dr. Mubashar Hassan

Department of Computer Science and Software Engineering
The University of Western Australia

Semester 2, 2024

Data Type: factors

- Tell R that a variable is nominal by making it a factor.
- The factor stores the nominal values as a vector of integers in the range `[ 1... k ]` (where `k` is the number of unique values in the nominal variable), and an internal vector of character strings (the original values) mapped to these integers.

```r
# variable gender with 20 "male" entries and
# 30 "female" entries
gender <- c(rep("male",20), rep("female", 30))
gender <- factor(gender)
```

Section 1

Bar Charts

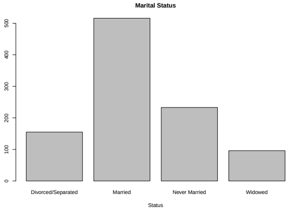
Bar charts (R)

A **bar chart** is a histogram for *discrete data*: it records the frequency of every value of a categorical variable.

But the basic R barplot requires the ‘table()‘ function to do the counting.

```
custdata <- read.table('../../data/custdata.tsv', header=T, sep='\t')
y <- table(custdata$marital.stat)   # table() carries out the aggregation
print(y)
##
## Divorced/Separated            Married      Never Married            Widowed
##                155                516                233                 96
```

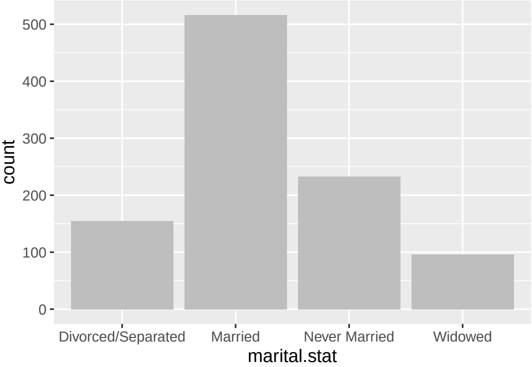
```
barplot(y, main="Marital Status", xlab="Status")
```



Bar charts (ggplot)

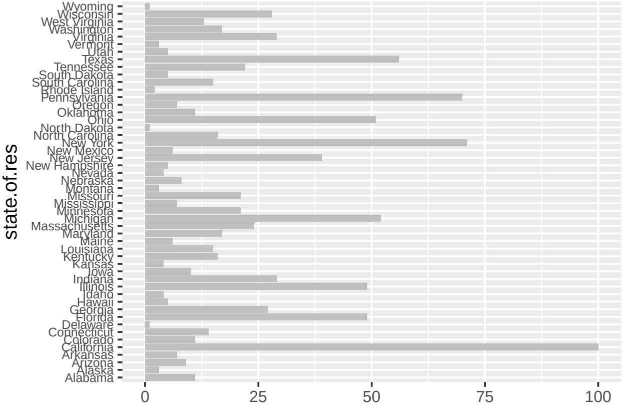
Using `ggplot()`, the code is simpler:

```
ggplot(custdata) +
  geom_bar(aes(x=marital.stat), fill="gray") +
  theme(text = element_text(size = 24))
```



Horizontal Bar Charts

```
ggplot(custdata) +
  geom_bar(aes(x=state.of.res), fill="gray") +
  coord_flip() +
  theme(axis.text.y=element_text(size=rel(0.8)))
```

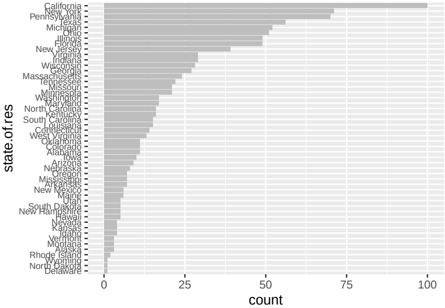


Bar charts are more informative if the data are sorted

```
# table() aggregates according to state.of.res
statesums <- table(custdata$state.of.res)
# as.data.frame() converts table object into a data frame
statef <- as.data.frame(statesums)
# define the column names for data frame statef
colnames(statef) <- c("state.of.res", "count")
# by default, order by statename alphabetically
summary(statef)
##      state.of.res     count
##  Alabama   : 1    Min.   :  1.00
##  Alaska    : 1    1st Qu.:  5.00
##  Arizona   : 1    Median : 12.00
##  Arkansas  : 1    Mean   : 20.00
##  California: 1    3rd Qu.: 26.25
##  Colorado  : 1    Max.   :100.00
##  (Other)   :44
```

Sorted Bar Plots

```r
statef <- transform(statef,
          state.of.res=reorder(state.of.res, count))
ggplot(statef)+
  geom_bar(aes(x=state.of.res,y=count), stat="identity",
           fill="gray") + coord_flip() +
  theme(axis.text.y=element_text(size=rel(0.8)))
```



Section 2

Dot Plots

Dot Plots are preferred by Cleveland

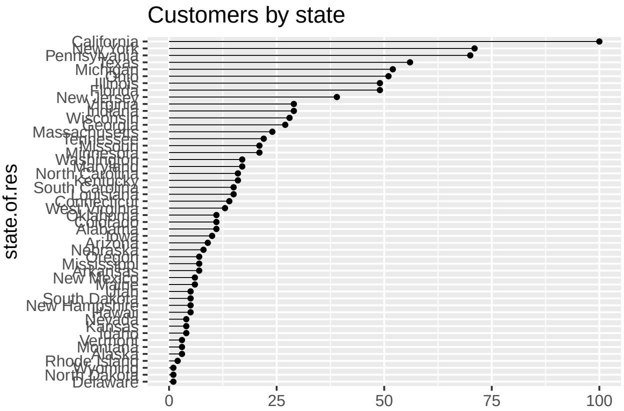
Cleveland prefers the dot plot to the bar chart for visualizing discrete counts.

- Bars are perceptually misleading.
  - Bars are two dimensional, a difference in counts looks like a difference in bar areas, rather than merely in bar heights.
  - the dot-and-line of a **dot plot** is not two dimensional, the viewer considers only the height difference when comparing two quantities, as they should.
- Bar charts or dot plot need to be sorted, to support more efficient extraction of insights.

(William S. Cleveland, The Elements of Graphing Data, Hobart Press, 1994.)

Dot plot in the WVPlots package

```
library(WVPlots)
ClevelandDotPlot(custdata, "state.of.res",
                 sort = 1, title="Customers by state") +
coord_flip()
```



References

- **Practical Data Science with R**, *Nina Zumel, John Mount*, Manning, 2nd Ed., 2020 (Chapter 3)
- **R for Data Science**, *Hadley Wickham, Garrett Grolemund*, O'Reilly, 2017 (Chapter 3)