CASSION · COURSE

Data Visualisation for Programmes

Charts that survive a printer, a projector and a sceptical reviewer, built in matplotlib and ggplot2 against the conventions each cluster already reads fluently.

Instructor	Pierre Robentz Cassion
Level	Intermediate
Learner hours	18 h
Taught in	Python · R
Updated	July 29, 2026
Sectors	Nutrition · Public Health · Food Security · Protection · WASH
Standards and methodologies	SMART survey · Integrated Food Security Phase Classification (IPC) · UNICEF indicator definitions · OECD DAC evaluation criteria · Core Humanitarian Standard (CHS)

Read and run the course online at data-analysis.cassion.dev/courses/data-visualisation

What you will be able to do

- Choose the mark from the comparison the chart is making, before opening a plotting library
- Put uncertainty on a chart in a form a programme audience reads correctly
- Use the sector's own colour conventions, and recognise the palettes that must not be repurposed
- Judge when a map is worse than a sorted bar chart, and use small multiples instead
- Produce a chart that survives greyscale, a projector and a colour-blind reader
- Label a figure so it stays correct when someone lifts it into another document

Prerequisites

- Applied Statistics for Programmes

Contents

About this course

A chart in a programme report is not decoration and it is not exploration. It is a claim, made in a form a reader absorbs faster than a sentence and questions less — which is exactly why it has to be built with more care than the sentence, not less.

This course teaches against the platform's own figure pipeline rather than alongside it. Every chart on this site is generated from a committed dataset by `scripts/figures/build.py`, in SVG for the web and TikZ for print, and nothing is drawn by hand. **Two of the figures this course uses were added by this course**, through that pipeline, which is the only demonstration of the argument that counts.

Three results shape it.

Four instruments, the same 1,955 households, and a seven-fold gap. Poor food consumption flags 7.4% and the coping index flags 51.0%. The food security course states those numbers in prose and a reader nods; the same four numbers as bars are not something anyone reads past.

A referral pathway loses more between consent and a referral being made than at any other gate. 88.5% of registered cases consent, 61.7% get a referral, 40.9% have it accepted. Written out it is a list; drawn, the largest drop is where nobody expects it, and it is a caseworker decision rather than a service failure.

The signal band is reserved and the reason is measurable. Green, amber and red mean normal, moderate and severe on this platform — a MUAC banding this audience reads without thinking. A chart that spends those three colours on three districts has taught the reader to stop reading them as a measurement, and no legend recovers it.

Both Python and R throughout, with matplotlib and ggplot2 as the working tools. You need *Applied Statistics for Programmes* first: half of good charting is knowing which number deserves an interval, and this course assumes you can already say.

Part I

The mark and the comparison

CHAPTER 1

The comparison decides the mark

Lesson 1 of 8 · 135 min

Four food security instruments on the same 1,955 households give 7.4%, 42.9%, 48.2% and 51.0%. Written as a sentence a reader nods; drawn as four bars nobody reads past it. The chart type was decided by the comparison, not by preference.

1.1 Start from the sentence, not the library

Before any plotting code, write the sentence the chart is making. The mark follows from it almost mechanically.

The sentence compares	Mark	Why
Categories against each other	**Sorted horizontal bar**	Length is the most accurately read encoding
One quantity over time	**Line**	Slope is the comparison; a bar chart of mon
A part against its whole	**A single bar, or the number**	A pie is two numbers and a legend
Two continuous variables	**Scatter**	Position on both axes
The same comparison across places	**Small multiples**	One shape repeated, read once
Where something stops	**Ordered stages**	The drop between bars is the finding

Nothing on this platform has yet needed a mark outside that list, which is a fact about programme reporting rather than about the list: the comparisons a programme makes are overwhelmingly categorical, temporal, or a proportion with an interval on it.

1.2 The four-instrument chart

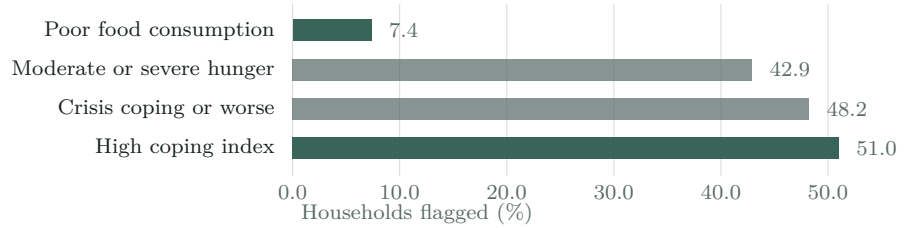

Figure 1.1: Four food security instruments applied to the same 1,955 households. The highest prevalence is seven times the lowest, so the instrument chosen decides the caseload before any analysis begins.

```
import pandas as pd
import matplotlib.pyplot as plt

prevalence = pd.Series({
    "Poor food consumption": 0.074,
    "Moderate or severe hunger": 0.429,
    "Crisis coping or worse": 0.482,
    "High coping index": 0.510,
```

```
})

fig, ax = plt.subplots(figsize=(7, 2.6))
ax.barh(prevalence.index, prevalence.values)
ax.invert_yaxis()                       # first row at the top
ax.set_xlabel("Households flagged (%)")
```

```
library(ggplot2)

ggplot(prevalence, aes(x = share, y = reorder(instrument, share))) +
  geom_col() +
  labs(x = "Households flagged (%)", y = NULL)
```

Horizontal, because the labels are words. "Moderate or severe hunger" rotated 45 degrees under a vertical bar is a chart that has decided its own labels are less important than its bars.

Sorted, unless the order means something. Alphabetical order is an ordering nobody asked about. Here the natural order is by value, and the sorted version makes the seven-fold gap the first thing seen.

One exception worth knowing. When the categories have their own order — IPC phases, MUAC bands, months, education grades — sorting by value destroys information the reader already has. **Sort by value, or by the category's own order, and never by the alphabet.**

1.3 The chart that is a table

```
coverage = pd.Series({"Measles first dose": 0.87, "Measles second dose": 0.61})
```

```
# Two numbers. A chart of two numbers is a chart of nothing.
```

Two or three numbers belong in a sentence. "Coverage falls from 87% to 61% between the first and second dose" is faster to read than any chart of it, occupies one line, and cannot be misread from the back of a room.

A chart earns its space at about five categories and stops being read at about twenty-five. Below five, write the sentence. Above twenty-five, you are drawing a table and should draw a table — or aggregate, which is a finding rather than a formatting choice.

1.4 Where something stops

A pathway is a sequence of stages, each a subset of the last, and the useful question is where the largest loss happens.

```
stages = pd.Series({
    "Registered": 1.000, "Consented to refer": 0.885, "Referral made": 0.617,
    "Referral accepted": 0.409, "Service reached": 0.388,
})
losses = -stages.diff().dropna()
print(losses.round(3))
print(f"largest drop: {losses.idxmax()} ({losses.max():.1%})")
```

```
# The drop is the finding; the bars are how you show it.
```

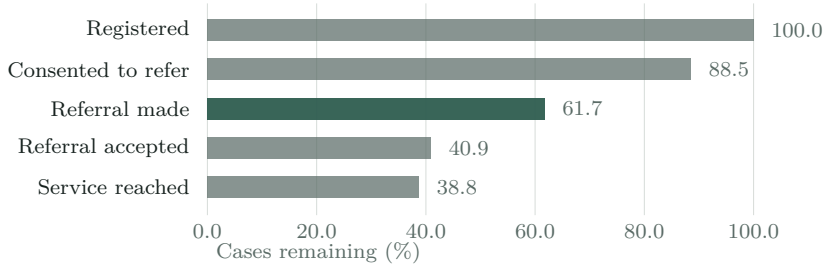

Figure 1.2: Protection referrals surviving each stage, as a share of the 1,850 cases registered. The largest single loss is between consent and a referral being made, which is a caseworker decision rather than a service failure.

The largest single drop is 26.8 points, between consent and a referral being made. Not at the service, not at acceptance — at the point where a caseworker decides whether to refer at all.

Bars in stage order, never sorted. The sequence is the meaning. And the figure marks the biggest drop, because a reader scanning five descending bars will otherwise take the last one as the story.

1.5 Three marks to avoid, and what to use instead

The pie chart. Angles are read less accurately than lengths, and a pie with more than three slices needs a legend to be read at all. **Use a sorted bar.** The one defensible pie is a single proportion against its complement, and a sentence beats that too.

The dual-axis chart. Two series on two scales, and the crossing point is decided by where you put the axes. **Use two small charts stacked**, sharing an x-axis, so a reader can see both without being told a relationship the scaling invented.

The 3-D anything. Depth adds no data and distorts every length it touches.

```
# The one line of matplotlib configuration this course insists on.
plt.rcParams.update({"axes.spines.top": False, "axes.spines.right": False})
```

```
theme_set(theme_minimal(base_size = 11))
```

1.6 The stacked bar, which is sometimes right

Stacking is the one contested case, so it is worth being precise.

Stacked bars are readable for the bottom segment and unreadable above it. Every segment except the first starts at a different baseline, so comparing the third segment across bars is comparing lengths that begin nowhere in particular.

Use stacking when the total matters and the composition is secondary — a caseload split by category, where the reader's first question is how big the caseload is. **Use grouped bars or small multiples when a specific component is the question.**

Never stack percentages that do not sum to 100. The four instruments above overlap: a household can be flagged by three of them. Stacking those four would draw a bar of 149.5% and imply a whole nobody has.

1.7 Report it whole

```
Food security instruments, 1,955 households, 2024

  Four instruments were applied to the same households and are shown as a
  sorted horizontal bar chart. The bars are prevalences, not components of a
  total: households can be flagged by more than one instrument, so the bars
  do not sum and are not stacked.

  Poor food consumption            7.4%
  Moderate or severe hunger       42.9%
  Crisis coping or worse          48.2%
  High coping index               51.0%

  The chart's claim is the spread, not any single bar. 76.3% of households
  are flagged by at least one instrument and 3.4% by all four.
```

The second paragraph is the one that stops the chart being misread, and it is two sentences. A reader who assumes those four bars are parts of something will build a caseload out of them.

1.8 What comes next

Every bar in this lesson is a proportion estimated from a sample, and not one of them carries an interval. The next lesson puts them on, and finds that the ordering of a twelve-commune chart survives it in only three cases.

CHAPTER 2

Twelve communes, one ranking, eleven overlapping intervals

Lesson 2 of 8 · 135 min

Sorted by prevalence the communes look like a league table. Every one of the eleven adjacent pairs has overlapping intervals, and only three clear the district median. A chart without intervals invites a ranking the data cannot support.

2.1 The ranking the chart invites

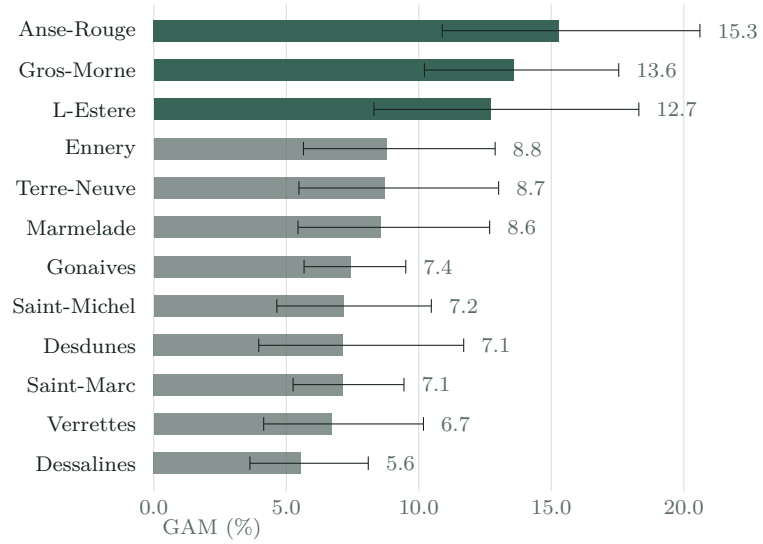

Figure 2.1: Global acute malnutrition by commune with 95% confidence intervals. The three highlighted communes are the only ones whose interval clears the district median.

Twelve communes, sorted, from 15.3% down to 5.6%. Without the whiskers this is a league table: someone is worst, someone is best, and a targeting decision follows.

```
import pandas as pd

communes = pd.DataFrame({
    "commune": ["Anse-Rouge", "Gros-Morne", "L-Estere", "Ennery", "Terre-Neuve",
                "Marmelade", "Gonaives", "Saint-Michel", "Desdunes",
                "Saint-Marc", "Verrettes", "Dessalines"],
    "gam":  [.153, .136, .127, .088, .087, .086, .074, .072, .071, .071, .067, .056],
    "low":  [.109, .102, .083, .056, .055, .054, .057, .046, .040, .053, .041, .036],
    "high": [.206, .175, .183, .129, .130, .127, .095, .105, .117, .094, .102, .081],
})

overlaps = sum(communes["high"][i] > communes["low"][i - 1]
               for i in range(1, len(communes)))
print(f"adjacent pairs whose intervals overlap: {overlaps} of {len(communes) - 1}")
```

```
# Eleven of eleven. The ranking is a ranking of point estimates only.
```

Every adjacent pair overlaps. The chart can support "these three are higher than the district median" and cannot support "Gros-Morne is worse than L'Estère".

The interval is what converts a ranking into a finding. Without it, the reader supplies an ordering the data does not contain, and the more carefully the bars are sorted the more convincing the invented ordering looks.

2.2 Drawing the interval

```
import matplotlib.pyplot as plt

fig, ax = plt.subplots(figsize=(7, 4))
ax.barh(communes["commune"], communes["gam"])
ax.errorbar(communes["gam"], communes["commune"],
            xerr=[communes["gam"] - communes["low"],
                  communes["high"] - communes["gam"]],
            fmt="none", capsize=3)
ax.invert_yaxis()
```

```
ggplot(communes, aes(x = gam, y = reorder(commune, gam))) +
  geom_col() +
  geom_errorbarh(aes(xmin = low, xmax = high), height = 0.3)
```

Four decisions, each of which changes what the reader concludes.

Say what the interval is. "95% confidence interval" in the caption, every time. A bare whisker is read as a standard error by some readers, a range by others, and a minimum and maximum by most.

Use the right interval. These are Clopper–Pearson exact binomial intervals, because several communes have small denominators and the Wald interval the statistics course rejected produces bounds below zero on exactly those cells.

Do not clip a bound to make it fit. An interval that reaches 20.6% sets the axis; truncating the axis to 15% and letting the whisker run off the edge is a chart that has hidden its own uncertainty.

Whiskers on top of bars, not instead of them. For a proportion the bar carries the magnitude and the whisker carries the doubt. A dot-and-whisker plot is equally defensible and reads better above about fifteen categories.

2.3 When intervals are not the right encoding

Three cases, and reaching for a whisker in any of them is worse than leaving it off.

A census, not a sample. A count of every water point in a district has no sampling error. It has other errors — a point missed, a status recorded wrongly — and a confidence interval does not describe them. **Say the denominator is complete instead.**

A total, not a rate. "1,850 cases registered" is a count of what happened. It carries no interval, and the protection course spent a lesson on why.

A time series a reader has to see the shape of. Twelve monthly points with whiskers becomes a hedge; a shaded band behind the line is the same information and leaves the shape legible.

```
ax.fill_between(months, lower, upper, alpha=0.2)
```

```
ax.plot(months, values)
```

```
geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.2) + geom_line()
```

2.4 The reference line does more work than the intervals

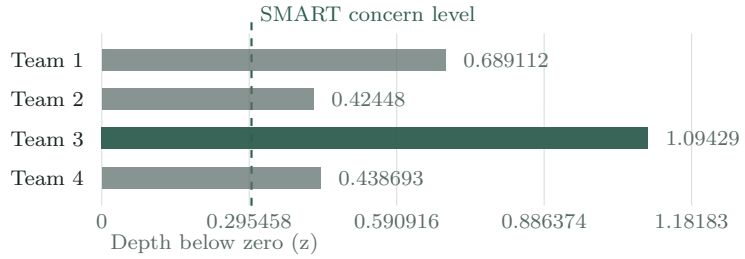

Figure 2.2: Weight-for-height mean by survey team, against the SMART concern level. One team's mean sits beyond the threshold at which SMART flags a measurement problem.

A number is compared against something whether or not you draw it, and drawing it decides what.

Sphere thresholds, SMART concern levels, IPC phase boundaries, national targets, a previous round. Each is a reference someone in the room is holding, and the chart that draws it is the chart everyone is looking at the same way.

Label the line with what it is, not with its value. "70% floor" beats "0.70", which the reader can already see on the axis.

One reference line. Two is a comparison the reader has to perform; three is decoration.

2.5 Reading order, and the three that clear

The figure marks three communes rather than describing them, which is a decision worth copying.

```
median = communes["gam"].median()
clears = communes[communes["low"] > median]
print(f"median {median:.1%}; clearing it: {list(clears['commune'])}")
```

```
communes |> filter(low > median(gam)) |> pull(commune)
```

Anse-Rouge, Gros-Morne and L'Estère are the only three whose lower bound exceeds the district median of 8.0%. That is a defensible statement and it is the one the highlight makes. Everything below them is a group, not a sequence.

Emphasis is an argument, so make it the argument the caption makes. Highlighting the top three by value would have marked the same three here by luck; highlighting by whether the interval clears the median marks them for a reason, and the two would diverge on a different dataset.

2.6 Report it whole

```
Global acute malnutrition by commune, Artibonite, 2024

  Prevalence by commune with 95% Clopper-Pearson intervals, sorted by point
  estimate. n ranges from 189 to 768 screenings per commune.

  Three communes -- Anse-Rouge, Gros-Morne and L'Estere -- have lower bounds
  above the district median of 8.0% and are highlighted.

  All eleven adjacent pairs have overlapping intervals. The chart supports
  "these three are above the district median" and does not support a ranking
  of one commune against its neighbour.

  Intervals are exact binomial. Communes with fewer than 300 screenings have
  intervals roughly twice as wide as Gonaives and Saint-Marc, which is a
  property of where the screening teams went rather than of their nutrition
  situation.
```

The third paragraph is what stops the chart being used as a league table, and it is the sentence a reader will not supply for themselves.

2.7 What comes next

Both figures in this lesson use one colour for the ordinary bars and one for the emphasised ones. The next lesson is about the three colours they deliberately do not use, and why spending them on districts costs something that cannot be bought back.

Part II

Conventions your reader already has

CHAPTER 3

The three colours you cannot spend twice

Lesson 3 of 8 · 135 min

Green, amber and red mean normal, moderate and severe to this audience before they read a legend. A chart that spends them on three districts has taught the reader to stop reading them as a measurement, and no legend buys that back.

3.1 Colour that already means something

This audience reads several palettes fluently, without a legend, because they are encoded in the standards they work to.

Palette	Means	Where it comes from
Green / yellow / red	Normal, moderate, severe	MUAC tape banding, triage
IPC phases 1–5	Minimal, stressed, crisis, emergency, catastrophe	IPC colour standard, fixed
JMP ladder	Safely managed → surface water	WHO/UNICEF joint monitoring
Blue	Water	WASH cluster convention

The IPC palette is not a suggestion. Phase 3 is a specific orange and a map that uses a different one for Phase 3 will be read as a different phase by someone who looks at IPC maps weekly. Use the published values or do not use phase colours.

The MUAC banding is why the signal palette is reserved on this platform. Green, amber and red are `-color-signal-normal`, `-moderate` and `-severe`, and they are for data state — indicator status, severity, quality flags — and nothing else.

3.2 What repurposing costs

```
# Three districts, three signal colours. Nothing here is a severity.
COLOURS = {"Nord-Ouest": "#2f8f5b", "Centre": "#d9a02b", "Sud-Est": "#c0442f"}
```

```
# A palette chosen because it is on the brand board, not because it means this.
```

A reader who sees red on Sud-Est reads "severe" before they read the axis. If Sud-Est happens to be the worst district the chart got lucky and taught a bad habit; if it is the best, the chart has said the opposite of its own data.

The cost is cumulative and it lands on the next chart. Once a reader has seen the signal band used decoratively, they stop reading it as a measurement — and the chart where it *is* a measurement has lost the shorthand it depended on.

So the rule is not aesthetic. Reserve the signal band for severity, quality flags and indicator status. Everything else gets the neutral palette.

```
INK, SPRUCE, SLATE = "#16211E", "#2F5D50", "#5A6B66"
ax.barh(labels, values, color=[SPRUCE if e else SLATE for e in emphasis])
```

```
scale_fill_manual(values = c("TRUE" = "#2F5D50", "FALSE" = "#5A6B66"))
```

Two colours is usually the whole palette a programme chart needs: one for the series, one for the thing you are pointing at. Every figure on this platform uses exactly that, and none has needed a third.

3.3 Colour as encoding, and what it can carry

Three uses, and only one of them is about which colour is prettier.

Categorical. Distinct hues, at most about six before a reader stops distinguishing them. Above six, use small multiples or direct labels instead of a legend.

Sequential. One hue, light to dark, for an ordered quantity. Coverage, rates, counts.

Diverging. Two hues from a meaningful midpoint — change from baseline, above and below a target. **The midpoint has to be real.** A diverging scale centred on the mean of the data invents a midpoint and colours half the districts red for being below average, which every dataset guarantees.

```
import matplotlib.pyplot as plt

fig, ax = plt.subplots()
ax.barh(districts, change, color=["#c0442f" if c < 0 else "#2f8f5b" for c in change])
ax.axvline(0, color="#16211E", linewidth=1)     # the midpoint, drawn
```

```
# If the midpoint is not drawn, the reader guesses where it is.
```

Here the signal colours are legitimate, because the encoding *is* a state: worse or better than the previous round, with zero as a real midpoint drawn on the chart.

3.4 Colour is the weakest channel; use position first

The ranking of visual encodings by how accurately they are read is not a matter of taste.

Rank	Encoding	Use for
1	Position on a common scale	The main comparison
2	Length	Bars
3	Angle, slope	Lines over time
4	Area	Rarely; a bubble is read as its radius
5	**Colour intensity**	An additional dimension, never the main one
6	Colour hue	Category labels

If the finding is only visible in the colour, the chart has encoded it in the weakest available channel. A choropleth is exactly this trap and the next lesson is about it.

Redundant encoding is not redundant. Marking the emphasised bars by colour *and* by a caption that names them means the chart still works in greyscale, at low resolution, and for a reader who cannot distinguish the two hues — which is lesson 5.

3.5 Nine ways sector charts get colour wrong

Each of these is common and each has a one-line fix.

Mistake	Fix
Signal colours on non-severity categories	Neutral palette
A rainbow scale for an ordered quantity	Single-hue sequential
IPC-like oranges that are not IPC values	Use the published values, or do not imply phases
Red for "female", blue for "male"	Neutral pair; sex is not a severity
Diverging scale centred on the data mean	Centre on a real midpoint, and draw it
A legend for two categories	Label them directly on the chart
Colour carrying the finding	Move the finding to position or length
More than six categorical hues	Small multiples
Colour and only colour distinguishing two lines	Add a direct label at the line's end

The last one is the most common in this sector, because a two-line chart looks finished. Print it in greyscale and it becomes one line and a mystery.

3.6 Report it whole

```
Figure conventions used in this report

  Charts use a neutral two-colour palette: one tone for the series, a darker
  tone for values the caption calls out.

  Green, amber and red are reserved for severity and data-quality status --
  MUAC banding, indicator status, completeness flags -- and are not used for
  geographic or demographic categories anywhere in this report.

  IPC phase maps use the published IPC colour values without modification.

  Where a chart uses a diverging scale, the midpoint is a stated reference
  (the previous round, or the national target) and is drawn on the chart.
```

Four lines in the methods annex, and they answer the question a reviewer asks about every map in the pack. A report whose colour conventions are stated is one whose charts can be trusted to mean the same thing twice.

3.7 What comes next

Colour intensity is the weakest encoding, and a choropleth puts the entire finding into it. The next lesson compares one map of twelve communes against twelve small charts, and finds the map cannot show the thing the last lesson established.

CHAPTER 4

The map cannot show the interval

Lesson 4 of 8 · 135 min

A choropleth of twelve communes puts the whole finding into colour intensity, which is the weakest encoding, and has nowhere to put the confidence interval that decided the finding. Twelve small charts show both. The map is right when the question is where, and only then.

4.1 What the map would have to leave out

The last two lessons established that three of twelve communes have GAM intervals clearing the district median, and that eleven of eleven adjacent pairs overlap. Draw that as a choropleth and ask what happened to it.

The finding	On a sorted bar chart	On a choropleth
Anse-Rouge is 15.3%	A bar length, read directly	A shade, read against a legend
Its interval runs 10.9–20.6	A whisker	**Nowhere**
Three communes clear the median	Three highlighted bars	A second map, or nothing
Eleven pairs overlap	Visible in the whiskers	**Nowhere**
Gonaives has 768 screenings, Desdunes 189	A note, or a second panel	**Nowhere**

A choropleth has one channel and the analysis produced four numbers per commune. Everything except the point estimate has to go somewhere else or be dropped, and in practice it is dropped.

And the one channel it has is the weakest. Colour intensity ranks fifth of six for how accurately a reader decodes it; a bar's length ranks second.

4.2 The area problem

```
areas = {"Gonaives": 502, "Anse-Rouge": 425, "Desdunes": 76, "Marmelade": 152}


# Visual weight is area. The finding is prevalence. These are unrelated.
```

A choropleth gives each commune visual weight proportional to its land area, which has nothing to do with its population, its caseload or its prevalence. A large sparsely-populated commune dominates a map that a bar chart would have put in the middle.

This is not a technicality in this sector. Rural communes are large and urban ones small, so a map of a health indicator systematically over-weights the rural cases and under-weights the place most of the children live.

4.3 When a map is the right chart

There are three, and they are worth being precise about because "we need a map" is usually a request for a map rather than a question about geography.

When the question is genuinely spatial. Where the flooding is. Which camps are within an hour of a functioning water point. Whether the unserved villages are contiguous or scattered — a question a bar chart cannot answer at all.

When the audience navigates by geography. A district health officer finds their own area on a map faster than in an alphabetical list, and that is a real cost saved.

When adjacency is the finding. Disease spread, market catchment, a cluster of non-functional water points along one road.

Otherwise, sort a bar chart. "Which communes are worst" is a ranking question, and a ranking is what a sorted bar chart is for.

4.4 Small multiples

The alternative to one map is twelve small charts, and it is stronger than it sounds.

```
import matplotlib.pyplot as plt

fig, axes = plt.subplots(3, 4, figsize=(10, 6), sharex=True, sharey=True)
for ax, (commune, series) in zip(axes.flat, monthly.groupby("commune")):
    ax.plot(series["month"], series["gam"])
    ax.set_title(commune, fontsize=9)
    ax.axhline(0.10, linestyle="--", linewidth=0.8)   # the same reference on each
```

```
ggplot(monthly, aes(month, gam)) +
  geom_line() +
  geom_hline(yintercept = 0.10, linetype = "dashed") +
  facet_wrap(~commune, ncol = 4)
```

Four rules, and breaking any of them turns twelve charts back into twelve unrelated charts.

Shared axes, always. If each panel scales to its own data the reader is comparing shapes drawn at twelve different magnifications, which is worse than no comparison.

Same reference line on every panel. It is what the eye travels along.

Order the panels by something. By value, by geography, by the order the reader will look for them. Alphabetical is the default and is almost never the useful one.

One shape, learned once. The reader decodes the first panel and then reads eleven more for free — which is the whole efficiency of the form and the reason it beats a map for a comparison question.

4.5 The chart a request for a map usually wants

Three substitutions that satisfy the underlying question at lower cost.

The request	What it usually wants	Give them
"A map of GAM by commune"	Which communes to target	Sorted bar chart with intervals
"A map of the water points"	How many work, and where the failures cluster	A bar chart by district, plus a ma
"A map of the caseload"	Where the people are	A map of *rates*, never counts

A choropleth of counts is a population map. It shows where people live, in every dataset, every time — because the count is the population times the rate and the population varies more. **Map rates, or map nothing.**

4.6 If you do draw the map

Four things it needs, and a map without them is worse than the bar chart it replaced.

Rates, not counts, for the reason above.

A stated class scheme. Quantiles, equal intervals, or the sector's own thresholds. Quantiles guarantee that a fifth of areas are in the worst class whatever the data says; equal intervals let one outlier flatten everything else. **IPC phases and MUAC bands are class schemes with meaning, and where they apply they beat both.**

The denominator on the page. A commune shaded dark on 189 screenings and one shaded dark on 768 are not the same claim, and the map cannot tell them apart.

A companion chart. The map answers *where*; the sorted bar chart with intervals answers *how much and how sure*. Publishing both costs one extra panel and is what lets the map be read safely.

4.7 Report it whole

```
Global acute malnutrition by commune: presentation

  Reported as a sorted bar chart with 95% intervals rather than as a
  choropleth. The finding is a ranking with substantial uncertainty, and a
  choropleth has one visual channel, which the point estimate occupies.

  Eleven of eleven adjacent communes have overlapping intervals; a map would
  show twelve distinct shades and imply twelve distinguishable levels.

  A commune map is included in Annex C for navigation, showing rates and not
  counts, with the number of screenings labelled on each commune and the
  same class breaks as the bar chart.
```

The last paragraph is the compromise that usually holds: the map exists, for the people who navigate by it, and it carries the denominators so it cannot be read as more than it is.

4.8 What comes next

Everything so far has assumed the reader sees the chart in colour, at full size, on a screen. The next lesson prints it in greyscale, projects it in a room with the lights on, and hands it to someone who cannot distinguish the two hues.

Part III

The page it must survive

CHAPTER 5

Print it in grey and see what is left

Lesson 5 of 8 · 135 min

A chart that works on your screen has been tested on the one display it will never be read on. The handout is greyscale, the projector washes out mid-tones, the room is lit, and about one man in twelve cannot separate your red from your green.

5.1 The four displays a chart has to survive

A programme chart is authored once and read on displays the author never sees.

Display	What it does	What fails first
A laser printer	Removes all hue	Two series distinguished only by colour
A projector in a lit room	Crushes mid-tones, lifts blacks	Light greys, thin lines, subtle shading
A phone on a metered connection	Shrinks everything	Type below 9 pt, dense axis labels
A colour-blind reader	Merges red and green	The signal palette, and any red/green pair

Every one of those is the normal case in this sector, not an accessibility edge case. The handout is photocopied, the projector is old, the reader is on a phone in a field office, and the room has a window.

5.2 The greyscale test, which takes ten seconds

```
import matplotlib.pyplot as plt

fig.savefig("chart.png", dpi=150)
grey = plt.imread("chart.png").mean(axis=2)     # crude luminance
plt.imsave("chart-grey.png", grey, cmap="gray")
```

```
# ggsave(), then open it in any viewer and desaturate. Or just print it.
```

Run it on every chart before it ships. If two series became one, the chart was encoding a distinction in hue alone.

The fix is never "pick better colours". It is to add a second channel:

Distinguish by	How
Two lines	**Direct labels at the line ends**, plus solid and dashed
Bars, one emphasised	**Lightness contrast**, not hue contrast
Categories	Small multiples, so nothing needs distinguishing
Points in a scatter	Shape as well as colour

Lightness survives greyscale by definition. The platform's figures use a dark spruce against a mid slate, which are two distinguishable greys before they are two colours — and that is why the emphasis in every figure on this site still reads in a photocopy.

5.3 Colour vision deficiency

About 8% of men and 0.5% of women of northern European descent have some form, and the commonest kinds merge red with green.

The signal palette is the problem case and it is unavoidable, because green, amber and red are the sector's convention and cannot simply be replaced.

```
SIGNAL = {"normal": "#2f8f5b", "moderate": "#d9a02b", "severe": "#c0442f"}
```

```
# Keep the hues. Add a second channel so the hue is not load-bearing.
```

Three ways to keep the convention and still be readable.

Order the categories. Normal, moderate, severe in that order means position carries the severity even when hue does not.

Label directly. The word "severe" on the bar costs nothing and removes the dependence entirely.

Vary the lightness deliberately. The three signal colours on this platform are not equally light: the green is darker than the amber, so they separate in greyscale as well as in hue. **Check that; a palette picked for hue alone often has two colours at identical lightness, which merge into one grey.**

5.4 Type, lines and the size the chart is actually read at

```
plt.rcParams.update({
    "font.size": 10,          # 9 pt is the floor for a printed figure
    "axes.labelsize": 10,
    "axes.titlesize": 11,
    "xtick.labelsize": 9,
    "lines.linewidth": 1.6,   # 1 pt disappears on a projector
})
fig.set_size_inches(6.5, 4)   # the width of a text column, not of a monitor
```

```
theme_set(theme_minimal(base_size = 11))
```

Author at the size it will be printed. A chart designed at 12 inches wide and dropped into a 6.5-inch column has half-size type, and the fix is not to scale the image — it is to redraw at the target width.

9 pt is the floor for print and 18 pt for a projected slide. The lesson decks this platform generates are 1280×720; anything below about 18 pt at that size is not read from the back of a room.

Thin lines and light greys are the first casualties of a projector. Use 1.5 pt or heavier, and do not rely on a grey lighter than about 40% for anything that carries information.

5.5 Everything that can be deleted

The strongest accessibility measure is having less on the chart.

Delete	Because
The legend, for two series	Direct labels are read in place
Gridlines, most of them	Four is enough; twelve is a texture
The chart border and top/right spines	They carry nothing
Redundant axis units on every tick	"%" on the axis label, not on each number
The title, when the caption says it	Two headings compete
Decimal places	42.9% not 42.94%; the interval is wider than the decimal

```
plt.rcParams.update({"axes.spines.top": False, "axes.spines.right": False})
ax.grid(axis="x", alpha=0.3)
ax.set_axisbelow(True)              # gridlines behind the bars
```

```
theme(panel.grid.minor = element_blank(), panel.border = element_blank())
```

Gridlines go behind the data. A gridline drawn over a bar reads as a division in the bar, which is a mark that means nothing.

5.6 Alt text, which is a caption doing a second job

Every figure on this platform carries a caption long enough to state its finding, and that caption is what a screen reader announces.

```
![Global acute malnutrition by commune with 95% confidence intervals. The three
highlighted communes are the only ones whose interval clears the district
median.](/figures/muac-gam-by-commune.en.svg)
```

"Chart of GAM by commune" describes the file, not the finding. A reader who cannot see the chart gets nothing from it.

A caption that states the finding serves four readers at once: the screen-reader user, the reader of the printed handout, the trainer working from the PowerPoint where the figure is a caption rather than a drawing, and the person skimming the report for the point.

This is why the platform requires it rather than recommending it. `figures.test.ts` fails a figure whose caption is shorter than twenty characters, and the reason is not tidiness.

5.7 Report it whole

```
Figure accessibility conventions

  All figures are readable in greyscale: emphasis is carried by lightness
  contrast rather than hue, and series are labelled directly rather than
  through a legend.

  Signal colours (green/amber/red) are used only where the categories are
  ordered by severity, so position carries the meaning independently of hue.

  Figures are drawn at the width they are printed. Minimum type size is 9 pt
  in the report and 18 pt on slides.

  Every figure carries a caption stating its finding, which is also its
  alternative text.
```

Four lines, and they are checkable by anyone with a printer. An accessibility statement that cannot be verified from the artefact is a claim rather than a convention.

5.8 What comes next

A figure is rarely read where it was written. The next lesson is about the caption that has to survive being pasted into a situation report by someone who did not do the analysis.

CHAPTER 6

The caption is the only part that travels

Lesson 6 of 8 · 135 min

Someone will paste your figure into a situation report without the paragraph that explained it. Whatever the caption says is what the figure will mean from then on, so the caption has to carry the finding, the denominator and the thing the chart cannot support.

6.1 Assume it will be lifted

A figure in a programme report has a second life. It is screenshotted into a situation report, pasted into a cluster presentation, forwarded as an image, and reproduced in a donor update — every time without the paragraph above it.

Write for the second life. The caption is the only part of the analysis that travels with the picture, so it has to be self-contained.

Caption
"GAM by commune"
"Figure 3"
"GAM by commune with 95% CIs. The three highlighted communes are the only ones whose interval clears the distr

6.2 Four things a caption has to carry

The finding, as a sentence. Not what the chart shows — what it *says*. "Counting improved sources overstates coverage in every district" is a finding; "Improved source against basic service by district" is a file name.

The denominator, or where to find it. 1,955 households, 1,850 cases, 24 schools. A percentage with no denominator is a number a reader cannot weigh.

The uncertainty, if the chart has any. "with 95% confidence intervals" is four words and it stops the chart being read as twelve exact values.

What the chart does not support. One clause, and it is the one that gets removed first when someone edits for length. "The bars do not sum" or "the comparison is observational" or "a ranking of one commune against its neighbour is not supported".

```
CAPTION = (
    "Four food security instruments applied to the same 1,955 households. "     #
    denominator
    "The highest prevalence is seven times the lowest, "                        # finding
    "so the instrument chosen decides the caseload before any analysis begins." #
    implication
)
```

```
labs(caption = str_wrap(CAPTION, 90))
```

6.3 The title and the caption do different jobs

A chart that has both should not say the same thing twice.

Element	Job	Length
Title	Names the comparison	3–6 words
Caption	States the finding and its limits	2–3 sentences
Axis label	Names the quantity and its unit	2–4 words
Direct label	Identifies a series in place	1–3 words

If the title states the finding, delete the caption or the title. Two headings competing is the commonest layout fault in a programme report, and the one that makes a page look busy without adding anything.

On this platform the title names and the caption tells. Every figure in `build.py` carries both, in both languages, and the split is enforced by the two fields being separate.

6.4 Labelling inside the chart

Four decisions that decide whether a reader has to look away from the data.

Direct labels beat legends. A legend makes the reader match a colour to a name and carry it back to the chart; a label at the end of the line does not.

```
for name, series in lines.items():
    ax.plot(series.index, series.values)
    ax.annotate(name, (series.index[-1], series.iloc[-1]),
                xytext=(6, 0), textcoords="offset points", va="center")
```

```
# ggrepel::geom_text_repel() on the last point of each series.
```

Label the values only when the reader needs them exactly. A bar chart with a value on every bar is a table wearing a chart's clothes. Label the emphasised bars, or none.

Never label a value to more precision than the interval justifies. 42.9% on a bar whose interval is ±3 points, not 42.94%.

Put the unit on the axis, once. "Households flagged (%)" and then bare numbers, not "%" repeated on every tick.

6.5 Numbers in a bilingual caption

This platform ships every figure in two languages, and the caption is where the two diverge most.

```
caption={
    "en": "Counting improved sources overstates coverage in every district. "
          "The difference is households whose source is improved and more than "
          "thirty minutes away.",
    "fr": "Compter les sources améliorées surestime la couverture dans chaque "
          "district. L'écart correspond aux ménages dont la source est améliorée "
          "et située à plus de trente minutes.",
}
```

```
# One figure, two captions, written together rather than translated later.
```

The decimal separator changes — 42,9 % in French, 42.9% in English — and the renderer handles it, which is why the figure carries values rather than formatted strings.

The space before the percent sign changes too, and French uses a narrow no-break space. Get it from the formatter rather than typing it.

Translate the finding, not the words. A French caption that reads as a translation has failed the same test as a French lesson that does.

6.6 The five-second test

Show the figure and its caption to someone who was not in the analysis, and ask them what it says.

If they describe the chart, the caption is a title. "It's a bar chart of communes" means the caption did no work.

If they state the finding, it works. "Three communes are clearly above the others" is the caption having done its job.

If they state a finding you did not make, fix the chart. A reader who says "so Dessalines is the safest" has read a ranking the intervals do not support, and the caption needs the clause that forecloses it.

6.7 Report it whole

```
Figure captions in this report

  Every figure caption states the finding as a sentence, names the
  denominator, and names one thing the figure does not support.

  Captions are written to be self-contained: a figure reproduced without its
  surrounding text should still carry its own interpretation and its own
  limits.

  Values are labelled to the precision the confidence interval supports and
  no further.

  Figures and captions exist in English and French. The captions are written
  in both languages rather than translated, so the finding rather than the
  wording is what is shared.
```

The second paragraph is the whole lesson, and it is a claim about the report's own robustness rather than about its style.

6.8 What comes next

Everything so far has been about what the chart says. The next lesson is about the axis, where a chart can say something different from its own data without a single number being wrong.

Part IV

Charts that cannot drift

CHAPTER 7

A gap of half a point, filling half the chart

Lesson 7 of 8 · 135 min

Boys attend 88.77% and girls 88.21%. On a zero-based axis that difference occupies 0.6% of the chart. Truncate the axis to 88–89 and it occupies 55.7%, a hundred-fold magnification, with every number still correct.

7.1 The same two numbers, four charts

The statistics course established that the attendance gap between boys and girls is 0.56 percentage points — significant at p = 0.023 on 68,267 marks, and equal to about a third of one school day per child per term.

```
import matplotlib.pyplot as plt

rates = {"Boys": 0.8877, "Girls": 0.8821}
gap = (rates["Boys"] - rates["Girls"]) * 100

for lo, hi in [(0, 100), (80, 95), (87.5, 89.5), (88, 89)]:
    share = gap / (hi - lo) * 100
    print(f"axis {lo}-{hi}: the gap fills {share:.1f}% of the chart height")
```

```
# Same data, four y-limits, four different stories.
```

Axis	The gap occupies	Magnification
0 to 100	0.6% of the chart	×1
80 to 95	3.7%	×7
87.5 to 89.5	27.8%	×50
88 to 89	**55.7%**	**×100**

Every one of those charts is accurate. The bars are in the right places, the axis is labelled, the numbers are correct. The last one shows a difference that no programme would act on filling more than half the plotting area.

Truncating a bar chart's axis is the single most effective way to mislead with correct data, and it is done far more often by accident than by design.

7.2 The rule, and the exception

A bar chart's value axis starts at zero. Always. A bar encodes its quantity as a length, and a length measured from a moving origin is not a quantity. This is not a convention with exceptions; it is what makes a bar readable.

A line chart's axis does not have to. A line encodes change as slope, and the reader is comparing points to each other rather than to the origin. A temperature series starting at zero degrees is a flat line and a wasted chart.

```
fig, (a, b) = plt.subplots(1, 2, figsize=(9, 3))
a.bar(rates.keys(), rates.values())         # must include zero
a.set_ylim(0, 1)
b.plot(months, completeness)                # need not include zero
b.set_ylim(0.25, 1.0)
```

```
# geom_col() + expand_limits(y = 0). geom_line() + coord_cartesian(ylim = ...).
```

And when you truncate a line's axis, say so on the axis. A break mark, or an axis label that starts visibly above zero, or a note. The reader who assumes zero and is wrong has read the chart backwards.

7.3 When the truncation is the honest choice

The completeness figure is the case, and it is worth being precise about why.

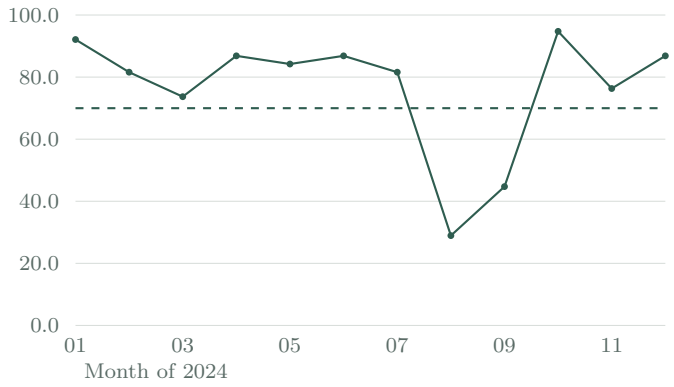

Figure 7.1: Monthly reporting completeness against the 70% floor. Completeness collapses to 28.9% in August and recovers to 94.7% in October, which is a reporting failure rather than a change in the epidemic.

```
values = [.921, .816, .737, .868, .842, .868, .816, .289, .447, .947, .763, .868]
print(f"range {min(values):.1%} to {max(values):.1%}")
```

```
range(values)
```

Completeness runs from 28.9% to 94.7% — a range of 65.8 points across a 100-point axis. The data uses two-thirds of the available height, so a zero-based axis wastes almost nothing and the shape is fully legible.

The rule to apply is: does the truncation change how large the variation looks? Here it barely would, so there is no argument for it. Where truncation would magnify a 0.5-point difference into half a chart, the answer is not to truncate more carefully — it is that a bar chart was the wrong mark and the number belongs in a sentence.

7.4 Five more axis decisions that change the reading

Reversed axes. A rate where lower is better — mortality, defaulting, stunting — drawn with the worst value at the top reads as a success. Say which direction is good in the axis label: "Defaulter rate (%), lower is better".

Log scales. Legitimate for an outbreak curve spanning three orders of magnitude, and misread by almost every non-technical audience as a linear one. **If you use one, label it and annotate a doubling.**

Percentage axes above 100. Gross enrolment reaches 109.2% in the education course's data, and an axis capped at 100 clips the bar and hides the finding.

Time axes with uneven spacing. Twelve monthly points drawn at even spacing when two months are missing draws a straight line through a gap. **Plot the dates, not the sequence.**

Aspect ratio. The same line at twice the height looks twice as steep. There is no neutral choice, so pick one and use it for every chart in the report, which at least makes them comparable to each other.

```
fig.set_size_inches(6.5, 4)      # every figure in the report, the same shape
```

```
# ggsave(width = 6.5, height = 4) as a project default.
```

7.5 The missing-data trap, which is an axis problem

The completeness figure exists because of this and it is worth stating directly.

A month with no reports drawn as zero is a claim that nothing happened. A month with no reports left out of the series entirely is a claim that time did not pass. Neither is true, and the honest chart shows the gap.

```
series = series.reindex(all_months)          # missing months become NaN
ax.plot(series.index, series.values)          # matplotlib breaks the line at NaN
```

```
# Do not tidyr::replace_na(0). A gap is data about the reporting system.
```

August's 28.9% is a reporting failure, not an epidemiological one. The epidemiology course spent a lesson on it, and the chart's job is to make it look like what it is — a hole in the denominator — rather than like a fall in cases.

7.6 Report it whole

```
Figure axis conventions

  Bar charts use zero-based value axes without exception. Where a difference
  is too small to be visible on a zero-based axis, it is reported as a
  sentence rather than as a chart.

  Line charts may use a truncated axis where the data range justifies it; the
  axis is labelled so the truncation is visible.

  Indicators where lower is better carry "lower is better" in the axis label.

  Months with no data are drawn as gaps, never as zero. Completeness is
  reported alongside any indicator whose denominator varies by month.

  All figures in this report are drawn at 6.5 by 4 inches, so vertical
  changes are comparable between charts.
```

The first paragraph is the one worth adopting verbatim. "Report it as a sentence rather than as a chart" is the decision that would have prevented the 88-to-89 chart, and it is a decision about honesty rather than about design.

7.7 What comes next

Every rule in this course is a rule someone has to remember on the day they make the chart. The last lesson removes the remembering, by generating the figure from the dataset — which is how every chart on this platform is built and why none of them can drift from the sentence beside it.

CHAPTER 8

The chart and the sentence come from the same file

Lesson 8 of 8 · 135 min

Every rule in this course is one someone has to remember on the day. Generating the figure from the dataset removes the remembering — and it is the only way a dataset correction moves the chart and the sentence beside it together, rather than one of them.

8.1 The failure this prevents

A chart is exported to PNG in March, pasted into a report, and the dataset is corrected in May. The prose is updated because someone reads it. **The chart is not, because nobody re-reads a picture.**

Nothing in any build can detect it. The figure is a valid image, the report is a valid document, and the number in the caption and the bar behind it now disagree.

The fix is structural rather than procedural. If the figure is generated from the dataset by a script that runs on every build, a correction moves both or neither, and "remember to regenerate the chart" stops being a thing anyone has to remember.

8.2 What this platform does

Every figure on this site is declared as a function that reads a committed CSV and returns a description of a chart.

```
def food_security_instruments() -> Figure:
    survey = read("food-security-survey-2024.v1.csv")
    coping = {row["household_id"]: row for row in read("livelihood-coping-2024.v1.csv")}

    # ... compute the four prevalences from the rows ...

    return Figure(
        slug="food-security-instruments",
        kind="bar",
        title={"en": "Four instruments, one set of households", "fr": "..."},
        caption={"en": "Four food security instruments applied to the same 1,955 "
                       "households. The highest prevalence is seven times the lowest...",
                 "fr": "..."},
        x_label={"en": "Households flagged (%)", "fr": "Ménages signalés (%)"},
        bars=bars,
    )
```

```
# The same shape in R: a function returning data plus its labels, rendered once.
```

No number is typed. The 7.4% in the caption is not written into the caption — the caption states the relationship and the bars carry the values, both computed from the file.

Two outputs from one description. SVG for the web, TikZ for the PDF handout and the Beamer deck. A figure written once appears in three places and cannot differ between them.

Standard library only. The generators need a stock Python and no install step, so a contributor can regenerate every figure without a toolchain.

8.3 The four properties worth copying

Whatever tool you use, these are what make the arrangement work.

One description, many renderings. The chart is data plus labels, and the renderer is separate. That is what lets the same figure be a web SVG, a print vector and a slide without three files that drift.

Bilingual by construction. Title, caption and axis label are dictionaries keyed by locale, so a figure cannot exist in one language. The platform's translation-parity rule reaches figures for the same reason it reaches lessons.

Committed output. The SVG and TikZ files are in the repository, so the site builds with no Python on the deploy path and a contributor without the toolchain can still ship a content change.

A test that fails when they diverge. `figures.test.ts` checks that every figure referenced by a lesson exists, that a French lesson references the French figure, that every figure exists in both languages and both formats, and that no caption is shorter than twenty characters.

```
# The check that matters most, in one line:
assert (OUT / f"{figure.slug}.{locale}.svg").exists()
```

```
# A test is what turns a convention into a constraint.
```

8.4 Doing it in matplotlib or ggplot2

The same discipline, without this platform's machinery.

```
import pathlib
import pandas as pd
import matplotlib.pyplot as plt

FIGURES = pathlib.Path("outputs/figures")

def gam_by_commune() -> None:
    screening = pd.read_csv("data/muac-screening-artibonite-2024.v1.csv")
    summary = (screening.assign(case=screening["muac_mm"] < 125)
               .groupby("commune")["case"].agg(["mean", "size"])
               .sort_values("mean"))

    fig, ax = plt.subplots(figsize=(6.5, 4))
    ax.barh(summary.index, summary["mean"])
    ax.set_xlabel("GAM (%)")
    fig.savefig(FIGURES / "gam-by-commune.svg", bbox_inches="tight")
    plt.close(fig)

    summary.to_csv(FIGURES / "gam-by-commune.csv")     # the numbers, beside the chart

if __name__ == "__main__":
    FIGURES.mkdir(parents=True, exist_ok=True)
    gam_by_commune()
```

```
gam_by_commune <- function() {
  summary <- screening |>
    mutate(case = muac_mm < 125) |>
    summarise(gam = mean(case), n = n(), .by = commune)

  ggsave("outputs/figures/gam-by-commune.svg", width = 6.5, height = 4)
  write_csv(summary, "outputs/figures/gam-by-commune.csv")
}
```

Write the numbers out beside the chart. A CSV next to every figure is what lets a reviewer check the chart without rerunning the analysis, and what lets the caption be verified against something.

One function per figure, one script that runs them all. Not a notebook where the chart is cell 34 and depends on cell 12 having been run.

Save as SVG, not PNG, wherever the destination allows it: it stays sharp when someone scales it, and it is text, so it diffs.

8.5 What to commit and what to ignore

Commit	Ignore
The generating script	The exported PNG previews
The figure output, if the build cannot run Python	Notebook checkpoints
The numbers CSV beside each figure	Anything with a timestamp in it

Deterministic output or nothing. A figure that embeds the generation date changes on every run, and the diff becomes noise that hides a real change. This platform pins `SOURCE_DATE_EPOCH` for exactly this reason, and the figures carry no date at all.

8.6 The three-line habit for a report with no pipeline

Most programme work does not have a figures pipeline and will not get one. The minimum viable version is still worth having.

One script that produces every figure in the report, run before the report is sent, in one command.

No chart made by hand in a spreadsheet, because that is the one that will not be regenerated.

The caption generated with the figure, or at least the numbers in it read from the same object the chart was drawn from.

```
caption = (f"Global acute malnutrition by commune, {summary['size'].sum():,} children "
           f"screened. {(summary['mean'] > 0.10).sum()} of {len(summary)} communes "
           f"are above the 10% emergency threshold.")
```

```
# If the caption is an f-string, it cannot disagree with the chart.
```

That single line is most of the benefit. A caption computed from the data is a caption that cannot survive the dataset being corrected without being corrected too.

8.7 Report it whole

```
Figure production for this report

  All figures are generated from the committed dataset files by
  scripts/figures.py, run as part of the report build. No figure is drawn by
  hand and no number in a caption is typed.

  Each figure writes its underlying values to a CSV of the same name, so any
  figure can be checked without rerunning the analysis.

  Figures are produced in English and French from one description, so a
  figure cannot exist in only one language.

  Output is deterministic: rerunning the script on unchanged data produces
  byte-identical files, so a diff shows only real changes.
```

The last paragraph is what makes the rest verifiable. A pipeline whose output churns on every run cannot be used to check anything, and one whose output is stable turns "is this chart current?" into a question the version control system answers.

8.8 What comes next

That is the course. Eight lessons and one idea: **a chart is a claim, and every decision in making it — the mark, the interval, the palette, the axis, the caption — is a decision about what the claim says.**

The lab builds two figures for a real report through a pipeline of your own, and the exercise takes a chart that is correct in every number and finds the four decisions that made it dishonest anyway.

About this edition

This PDF is generated from the same source as the web version of the course, so the two cannot drift. The web edition carries the runnable code, the downloadable datasets and any corrections issued since this file was produced.

> Every dataset referenced in this course is synthetic or fully de-identified. No record traces to a real person.

This course is published in English and in French. The French edition is a professional adaptation using the terminology UNICEF, WHO, WFP, OCHA and national ministries publish in — not a literal translation.

Read and run the course online at data-analysis.cassion.dev/courses/data-visualisation

Generated July 29, 2026.