Visualization and Data Presentation

Tomáš Kalibera, Vlastimil Babka
Example: SPEC JVM 98 Compress Benchmark

- _201_compress
  - LZW compression of real files (always the same ones)
  - Reports execution time
  - 30 executions of each version (below), omitting first 5 values as warm-up

- 32 bit vs. 64 bit JVM (x86_64 CPU)
  - HotSpot "server" implementation
  - "-d32" for 32bit, "-d64" for 64bit
  - Comparing the results
Run-Sequence Plot (SPEC Compress Example)
Box-Plot (SPEC Compress Example)
Box-Plot (SPEC Compress Example)

- **Outlier**: A point farther from the median than the whisker.
- **3rd quartile**: Drawn at 1.5*IQR from the 1st quartile.
- **Median**: The middle value separating the higher half from the lower half of the data.
- **Hinge**: Drawn at the 1st quartile.
- **Whisker**: Drawn at the farthest data point from the box, which is not farther than 1.5*IQR from the 1st quartile.
Box-Plot (SPEC Compress Example)

notch

non-overlapping notches show that medians for two data sets are different; this comparison should correspond to checking if a 95% confidence interval for the median difference includes zero.

width = \(1.58 \text{IQR}/\sqrt{n}\)

assumes similar sample set sizes, asymptotically normal median
Many variants exist

- Which differ in where/if whiskers are plotted, how far are hinges from the median, if outliers are plotted, etc.
- So check the software you use and report the definition with your results

In R

- The default (plus notch) as shown in the example
- Function "boxplot"

```r
boxplot(list(d32, d64), names=c("32-bit", "64-bit"), notch=TRUE, range=1.5, varwidth=TRUE)
```
Box-Plot Use

- **One-sample**
  - Location
  - Variability, Skew
  - Outliers

- **Multi-sample**
  - Comparison of location/median
  - Does location differ depending on parameter? (have boxplots for different parameter values)
  - Does variability differ depending on parameter? (have boxplots for different parameter values)
Alternative: Violinplot

- Combination of box plot and density plot

- require ("vioplot")
- vioplot (v1, v2, ...)

Vlastimil Babka
Bihistogram – from NIST Handbook

- Does a (2-level) factor influence results?
  - Location, variability, mean, distribution shape
- Are there outliers?
- In R
  - Hmisc package, function "histbackback"
  - Manual plotting using other functions in R might give better results, i.e. “barplot”
So far we mostly covered visualization methods for exploring the data
- That is for "us" to aid in evaluation of results

But we also need to present results of the analysis to other people
- To "decision makers"
- Reviewers of scientific papers
- The public, etc.

Convey the data effectively, accessibly

Avoid “Pictorial Games”
- Learn to recognize them
Visual Data Presentation Guidelines

- Easy to understand with minimum effort
  - Don't use overly complex methods
    - Plots should make understanding the data simpler
  - Use descriptive labels as opposed to abbreviations
  - Simplify terms based on the knowledge of target audience
  - Each graph should have one (or not too many) messages
  - Don't show unrelated things in a single plot (i.e. to save space)
Use Descriptive Labels, Not Symbols

(a) Symbols

Response time

(b) Key words

Figures: [Jain]
Presenting Too Much in Single Chart

Figure: [Jain]
Visual Data Presentation Guidelines

- **Self-contained**
  - Always include all information needed to understand the plot into the plot
    - Many people do not read the text or pay attention to anything but pictures
    - Some people only read the text if they like the pictures

- **Fair**
  - Simplification can be done in many ways which can make your claims stronger or weaker; it is nice not to abuse this
Base Rules (What People Expect)

- Always label axes
  - All axes in all pictures at all times
- Always include units, type of scale
  - On all axes in all pictures at all times
- X-Axis is independent variable
  - Observation index, measurement date, ...
- Y-Axis is dependent (random) variable
  - Execution time, throughput, ...
- Values grow from left to right, bottom to top
- Clearly distinguish what is measured from what comes from workload definition
Base Rules

- Repeatability
  - If your experiment is restarted or data re-sampled, it should lead to very similar graphs.
  - Show confidence intervals, whenever a random quantity is represented by a single number/point.

- Conciseness
  - Don’t show what is not needed to convey your message, this is typically:
    - Double axes, grids (or too dense grids)
    - Irrelevant factors

- Minimize Ink – Maximize information/ink ratio
Pictorial Games – No Confidence Intervals

(a) Without confidence interval

(b) With confidence interval

Figures: [Jain]
Minimize Ink

Figures: [Jain]
More Guidelines (Follow if Possible) [Jain]

- **Ranges**
  - Include (0, 0), which is bottom left
  - Show numbers for the minimum and maximum value shown in the graph
  - *Choosing range on Y-axis makes a difference look bigger/smaller*
    - Three-quarter-high rule:
      "vertical height of the top-most point is at least three-quarters of the horizontal offset of the right-most point"
- **What about whole plot’s aspect ratio?** (Tufte)
  - About 50% taller than wider, Golden Section (1.618) common
Pictorial Games – Non-Zero Origins; Range

Figures: [Jain]
More Guidelines (Follow if Possible)

- **Scales**
  - Linear
    - avoid logarithmic if possible, if not, mark logarithmic axes very clearly
  - Continuous
    - Broken axes, if needed, should be marked significantly
    - Broken axes make sense for clustered data
    - If an axis is broken, the different parts should at least have the same units
    - *Breaking Y-axis (i.e. in bar plots) makes differences look bigger than with a continuous axis*

- **Rounding rules**
  - Rounding rules for numeric results apply to labels
Pictorial Games – Broken Scales

Resp. Time

System

Resp. Time

System

Figures: [Jain]
More Guidelines (Follow if Possible)

- Labeling of curves instead of legend boxes
  - Especially with many curves in a plot, it might be annoying having to identify which curve corresponds to which label in the legend
  - Jain thus suggests to label individual curves in the graph
    - Still not good to have more than 6 curves

- However...
  - Legend boxes are quite common
  - Legend boxes are easier to implement
    - Though manual intervention is still often needed to avoid overlap with graph
Example: Legends vs Direct Labels

(a) Direct labeling

(b) Legend box

Figures: [Jain]
More Guidelines (Follow if Possible)

• Ordering of qualitative data
  - Should be consistent (use one of following)
  - Should be systematic (use one of following)

• Common methods
  - Alphabetic using labels
  - Best to worst, worst to best
  - Original (base), our improvement
  - Time of publishing, ...
Avoiding Visual Tricks

- Histogram bin size
  - Selection of bin size should lead to a stable, repeatable histogram
    - Re-sampling does not change it significantly
    - Very small changes in bin size do not change it significantly
    - Re-starting the experiment does not change it significantly
  - Bin size can be abused to make the histogram seem similar to particular distribution (Normal, Exponential, ...)

Tomáš Kalibera
Pictorial Games: Histogram Buckets

Figures: [Jain]
Avoiding Visual Tricks [Jain]

- Pictograms size
  - If pictograms are used to show a quantity, they should be scaled by the area
  - *If scaled by length, they make differences look bigger*
  - Use proper scaling factor, or better, do not use 2D area for 1D data at all
    - Num. of D’s to show the data should not exceed the D’s in the data

.. 4x faster, not just 2x
Avoid dramatizing results by showing redundant (inverse) information
- This is sometimes done with throughput and response time
- The trend just looks bigger...
Avoiding Tricks – Omitting Context

- Showing only the part of data that favors us

Model precision validated only for small number of clients

The measured behavior might actually look like this for more clients

Pictures from our EPEW’07 slides.
Note: they do not depict the same system
Keeping Plots Comprehensible [Jain]

- Don't present too many alternatives in one plot
  - "5-7 messages at a time"
  - Line charts with max 6 curves
  - Column chart with max 10 bars
  - Pie chart with max 8 components (if at all)
  - Each cell in a histogram should have at least 5 points in it
- Don't use too many Y-variables
- Don't use line chart for qualitative data
  - Connecting two points usually suggest that intermediate values can be interpolated
Bad Use Of Line Chart

Figures: [Jain]
More (Advanced) Guidelines (E. Tufte)

  - Not specific for performance results
  - Some similar to those already mentioned

- Do not use graphics in all cases
  - Small data sets, trivial data
  - Non-comparative, highly labeled data
  - Tables might be better
  - Sentences for no more than two numbers
More (Advanced) Guidelines (E. Tufte)

- Above else show the data
- Maximize data-ink ratio (within reason)
  - Erase non-data ink
    - *Thick grids, frequent tick marks*
  - Erase redundant data ink
More (Advanced) Guidelines (E. Tufte)

- Avoid chartjunk
  - Moire patterns (for shading barplots, etc.)
  - Self-promoting graphics (“Ducks”)
Maximizing data-ink by erasing

- Box-plot – *quartile plot*
More (Advanced) Guidelines (E. Tufte)

- Maximizing data-ink by erasing
  - Barplot – *white grid*
Maximizing data-ink by erasing

- Scatterplot – range-frame
  - Also good for run-sequence plot (Y-axis only)

With labels:
• Maximizing data-ink by erasing
  ▪ Scatterplot – *dot-dash-plot*
More (Advanced) Guidelines (E. Tufte)

- Avoid puzzles
  - Example: color combinations
  - Color ordering also not intuitive (shades of gray better)

- If more dimensions needed, consider **small multiples**
  - Single understanding of multiple repeated plots
Accessibility – *friendly data graphics* (E. Tufte)

- Avoid abbreviations
- Use horizontal labels (also for Y-axis)
- Use little messages to explain data
- Elaborately encoded shading (no colors), direct labels instead of legend
- Graphics attractive, provokes curiosity, no chartjunk
- If colors used, remember color-blind viewers
- Clear, precise, modest typing, upper-and-lower case, with serifs
Other Common Visualization Techniques..
Normalization – Altering scale and/or variance

\[ E[aX - b] = aE[X] - b \]

\[ \text{var}[aX + b] = a^2 \text{var}[X] \]

\[ \text{var}\left[ \frac{X - E[X]}{\sqrt{\text{var}[X]}} \right] = 1 \]

\[ E\left[ \frac{X - E[X]}{\sqrt{\text{var}[X]}} \right] = 0 \]
Normalization

- Both mean and variance
  - Getting 0 mean and 1 variance
  - Allows comparing trends / checking dependencies of variables with different location / scale / units
  - Called "normalization" by statisticians

- Providing relative values
  - Just dividing by a base / known value, i.e. results we want to improve
  - Called "normalization" by non-statistician practitioners, including CS
Normalization Summary

- Make sure to state clearly what kind of normalization you use
  - "normalized against" or "normalized to have zero mean and unit variance" or something along these lines
  - Don't use statistical normalization to audience that cannot get what it means..
  - *In case of normalizing against something, "ratio games" apply*
## Case Study 11.1: 6502 vs. 8080

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>System</th>
<th>6502</th>
<th>8080</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block</td>
<td>41.16</td>
<td>51.50</td>
<td></td>
</tr>
<tr>
<td>Sieve</td>
<td>63.17</td>
<td>48.08</td>
<td></td>
</tr>
<tr>
<td>Sum</td>
<td>104.33</td>
<td>99.58</td>
<td></td>
</tr>
<tr>
<td>Avg</td>
<td>52.17</td>
<td>49.79</td>
<td></td>
</tr>
</tbody>
</table>

Which system is faster?
# Case Study 11.1: 6502 vs. 8080

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>System</th>
<th>6502</th>
<th>8080</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block</td>
<td>41.16</td>
<td>51.50</td>
<td></td>
</tr>
<tr>
<td>Sieve</td>
<td>63.17</td>
<td>48.08</td>
<td></td>
</tr>
<tr>
<td>Sum</td>
<td>104.33</td>
<td>99.58</td>
<td></td>
</tr>
<tr>
<td>Avg</td>
<td>52.17</td>
<td>49.79</td>
<td></td>
</tr>
</tbody>
</table>

1. Ratio of Totals

- Conclusion: 6502 is worse. It takes 4.7% more time than 8080.
### 2. 6502 as the base:

<table>
<thead>
<tr>
<th>System</th>
<th>6502</th>
<th>8080</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>1.25</td>
<td>1.00</td>
</tr>
<tr>
<td>1.00</td>
<td>0.76</td>
<td>1.31</td>
</tr>
<tr>
<td>2.00</td>
<td>2.01</td>
<td>2.11</td>
</tr>
<tr>
<td>1.00</td>
<td>1.01</td>
<td>1.06</td>
</tr>
</tbody>
</table>

### 3. 8080 as the base:

<table>
<thead>
<tr>
<th>System</th>
<th>6502</th>
<th>8080</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.80</td>
<td>1.00</td>
<td>1.31</td>
</tr>
<tr>
<td>1.00</td>
<td>1.00</td>
<td>2.00</td>
</tr>
<tr>
<td>1.06</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

1. Ratio of Totals: 6502 is worse. It takes 4.7% more time than 8080.
2. With 6502 as a base: 6502 is better. It takes 1% less time than 8080.
3. With 8080 as a base: 6502 is worse. It takes 6% more time.
Barplot

- Showing more dimensions
  - Usually one continuous variable (execution time, ...)
  - And two qualitative variables – two factors
Barplot: Example

From OVM paper by Baker, Cunei, Kalibera, Pizlo, Vitek
Barplot: Example

- Evaluation of different GC techniques and JVMs
  - GCJ, HotSpot (1.5,1.6, client and server), OVM (Henderson and Thunked GC)
- Using different SPEC JVM 98 benchmarks
  - Compress, db, jack, javac, jess, ...
How fast can you tell how is the data normalized?
(it is indeed stated explicitly in the plot's caption in the paper)
Barplot Example: Same Results, Normalized

Execution time relative to HS-1.5-Clt

- GCJ-4.0.2
- HS-1.5-Clt
- HS-1.5-Srv
- HS-1.6-Clt
- HS-1.6-Srv
- Henderson
- Thunked

Tomáš Kalibera
Notice how some differences are magnified or reduced.
Barplot Example: Normalized vs. Absolute Values

- normalized values
- absolute values

Comparison of execution times for different programs (compress, db, jack, javac, jess, mpegaudio, mtrt) and different configurations (GCJ-4.0.2, HS-1.5-Clt, HS-1.5-Srv, HS-1.6-Clt, HS-1.6-Srv, Henderson, Thunked).

Execution time [s]

Programs:
- compress
- db
- jack
- javac
- jess
- mpegaudio
- mtrt
- g. mean

Configurations:
- GCJ-4.0.2
- HS-1.5-Clt
- HS-1.5-Srv
- HS-1.6-Clt
- HS-1.6-Srv
- Henderson
- Thunked
Barplot

- Implementation
  - In R, function "barplot"
  - Input is a matrix
    - Columns are the groups of bars
    - Rows are the individual bars in each group

  ```
  barplot(my_matrix, beside=TRUE)
  ```

- Data ordering
  - Might encode some extra information in the data
  - There is nearly always better ordering than alphabetical (Tufte, on tables)
Barplot – Ordering of Data

Benchmarks ordered by their decreasing isolated cache miss rate

Figure 2. Slowdown of SPEC CPU2006 benchmarks due to sharing, AMD Server
Gantt Plot

- Disambiguation
  - "Project management" version of the graph
    - X-axis – project timeline
    - Y-axis – tasks, with dependencies
  - "OS Scheduling" version
    - Similar to project management, X is timeline, Y is tasks
  - "Performance evaluation" version
    - X-axis is NOT timeline (it's harder to explain what it actually is)
    - Y-axis is resources (CPU, I/O, ...)

Tomáš Kalibera
Gantt Plot – Project Management

WBS 1 Summary Element 1
- WBS 1.1 Activity A: 75% complete
- WBS 1.2 Activity B: 67% complete
- WBS 1.3 Activity C: 50% complete
- WBS 1.4 Activity D: 0% complete

WBS 2 Summary Element 2
- WBS 2.1 Activity E: 0% complete
- WBS 2.2 Activity F: 0% complete
- WBS 2.3 Activity G: 0% complete

Source:
http://en.wikipedia.org/wiki/File:GanttChartAnatomy.png
Gantt Charts

- Shows relative duration of a number of conditions

![Gantt Chart Diagram](image)
Gantt Plots for Resource Utilization

- Motivation
  - All resources should have reasonable utilization
    - Too high – potential bottleneck of the system
    - Too low – they are not used efficiently, we might have saved money if they were worse
  - For workloads / benchmarks
    - Should put the system into state when different combinations of resources are utilized at different times

... which can be visualized by Gantt graphs
Kiviat Plot: Names and Variants

- Kiviat Plot
- Star Plot
- Radar Plot
- Spider Plot
Kiviat Plot Variants

- Any multivariate data
  - Not necessarily alternating LB/HB metrics, can be all LB or all HB
- Multiple measurements in a single plot
  - Area not filled-in
  - Resembles spider web and spider
    - The spider web can be shown instead of tick marks on the spokes
Star Plot Example (NASA)

All metrics are LB – Closer to the center is better

http://start1.jpl.nasa.gov/images/MERStarPlot.gif
Star Plot Example (NASA)

http://start1.jpl.nasa.gov/images/MERStarPlot.gif
Kiviat Plot Example (Benchmarking)

- SPECjvm95+DaCapo benchmarks (small, large data sets)
- Five different JVM’s
- 30+ hardware perf. counters measured
  - Principal Component Analysis (PCA) used to pick smaller set of most important counter combinations
- Questions w.r.t. benchmarks-JVM interactions
  - Are the benchmarks representative enough?
  - Are the JVM’s significantly different?
- Use Kiviat plots of the PCA components

The idea and following plots are from talk by Andy Georges (link on last slide)
pmd and fop have very similar shapes (also bloat)
large differences between JVM’s

<table>
<thead>
<tr>
<th>antlr</th>
<th>pmd</th>
</tr>
</thead>
<tbody>
<tr>
<td>lackdown</td>
<td></td>
</tr>
<tr>
<td>IBM J9</td>
<td></td>
</tr>
<tr>
<td>.kes RVM</td>
<td></td>
</tr>
<tr>
<td>JRockit</td>
<td></td>
</tr>
<tr>
<td>SUN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>antlr</td>
</tr>
<tr>
<td>----------------</td>
<td>-------</td>
</tr>
<tr>
<td>Blackdown</td>
<td>★</td>
</tr>
<tr>
<td>IBM J9</td>
<td>★</td>
</tr>
<tr>
<td>Jikes RVM</td>
<td>★</td>
</tr>
<tr>
<td>JRockit</td>
<td></td>
</tr>
<tr>
<td>SUN</td>
<td></td>
</tr>
</tbody>
</table>
no more workload redundancy, still JVM differences
Kiviat Plot and Variants Summary

- Presentation of relative multivariate values
  - Of not too many measurements (single plot) or more using *small multiples*
  - Good for shape comparison
- Predefined sets of metrics [Jain]
  - Kiviat graphs for CPU/IO utilization
  - Kiviat graphs for network utilization
- In R
  - "stars" function
Links

  - Chapter 10 (up to 10.6.2)
  - 1.3.3. Graphical Techniques: Alphabetic
- Andy Georges: Evaluation in Computer Science (slides from a D3S seminar talk)