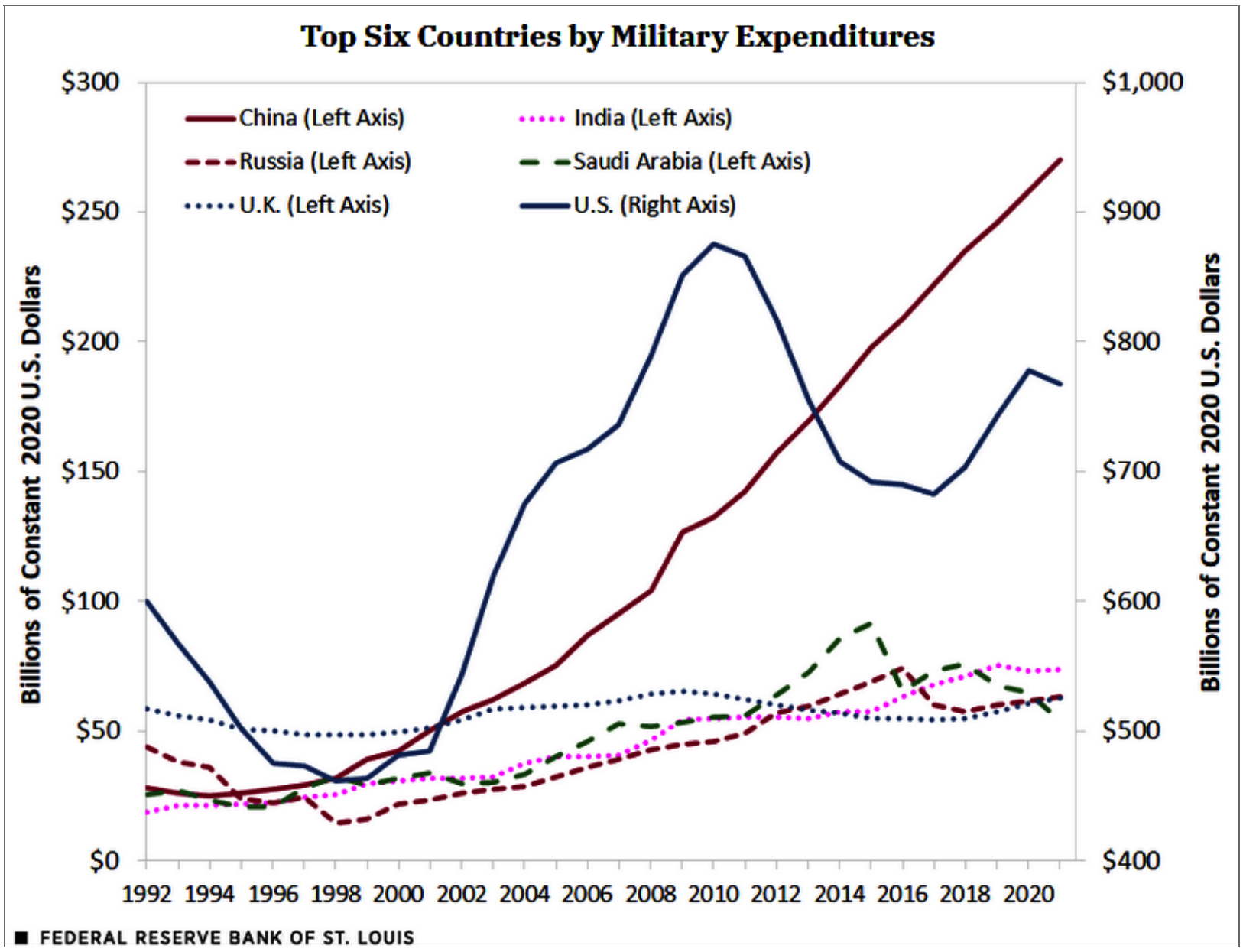
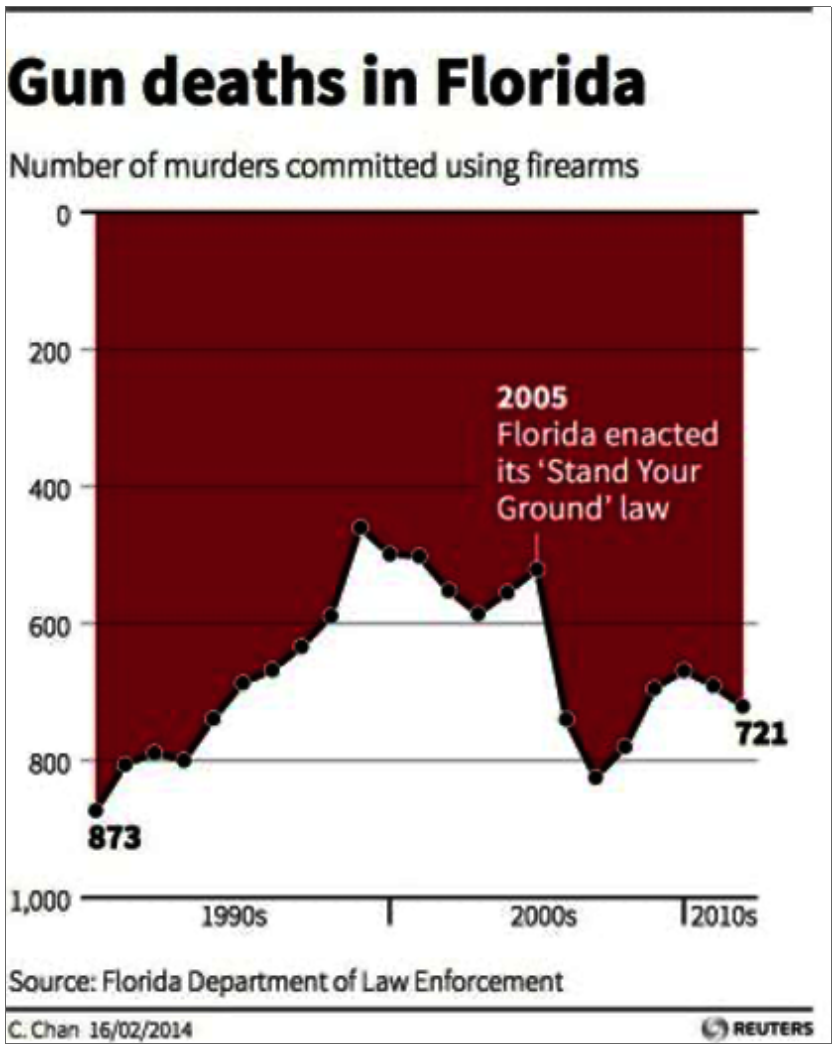
Lecture 2: Data Abstraction and Representation

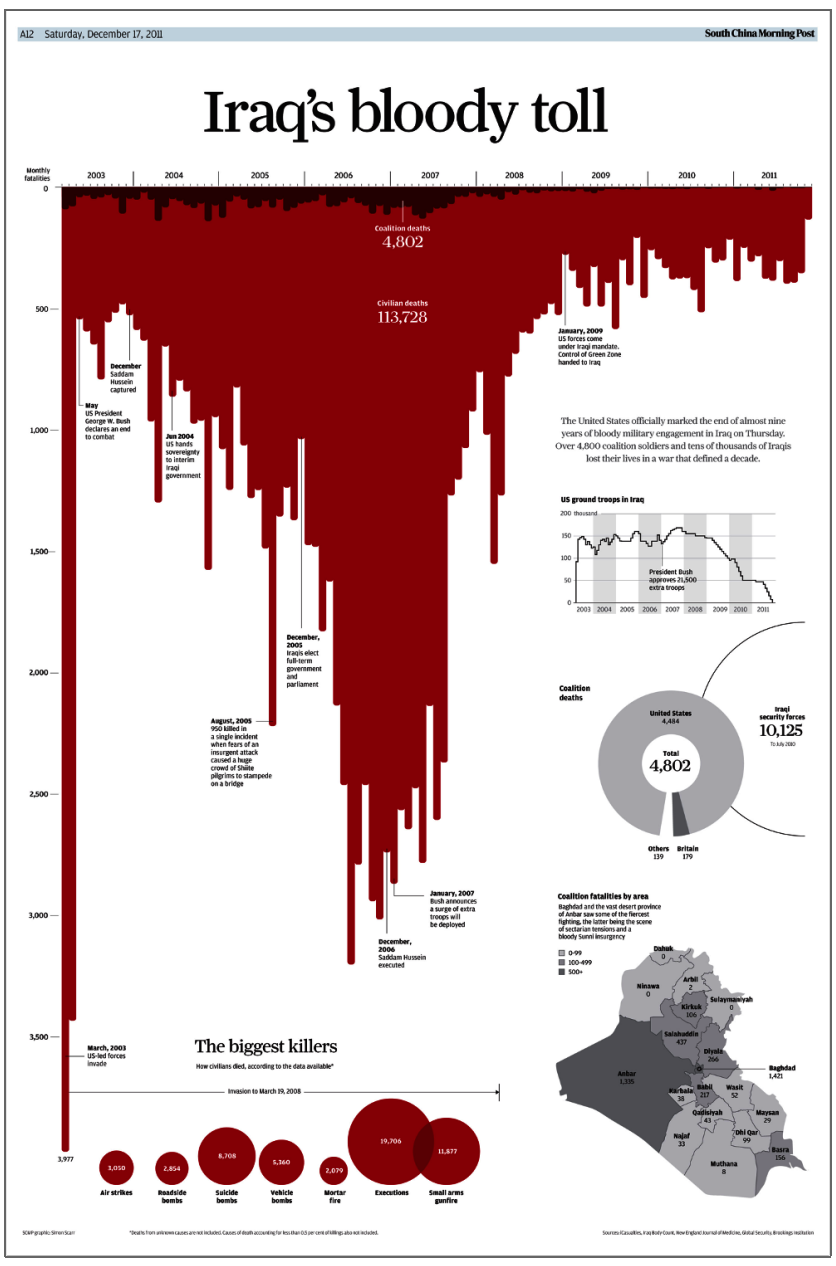
Today's Visualization

Today's visualization is *not* chosen for skillful or appropriate results, but for the somewhat duplicitous use of scales to communicate something *other than the data*.
Task: Describe, as completely as you can, the design of this graph. Pay attention to whether distinctions are drawn between parts of the data.
Question: Why did I call this *duplicitous*? What do you think the editorial intent was?



Catching up: Gun Deaths in Florida





Question: Was the intended metaphor successful? In "Gun deaths in Florida"? In "Iraq's bloody toll"? What could have been done differently to make the message more efficiently conveyed?

Catching up: Your First Aesthetic Critique

What differences and similarities do you see between the different “Out of the box” plots here?
What would you like to change?
What would you like to check / verify?
Would you like more (or less) binning and aggregation?
What, if any, interactive features would you like?
What, if any, labels, titles, annotations would you like to use?
What would an interesting use case for this plot be?
If you were to pull properties and features freely from all platforms (or add yourself) – how could you specify the most appropriate plot for this use case?

Catching up: Your First Aesthetic Critique

Garima Goyal

Catching up: Your First Aesthetic Critique

Mahfal Naleemul Rahuman

Catching up: Your First Aesthetic Critique

Larry Ryan

Catching up: Your First Aesthetic Critique

Larry Ryan

Catching up: Your First Aesthetic Critique

Sean Sudol

Catching up: Your First Aesthetic Critique

Ryan Mc Neil

Catching up: Your First Aesthetic Critique

GiBeom Park

Catching up: Your First Aesthetic Critique

Joshua Rollins

Catching up: Your First Aesthetic Critique

Giacomo Radaelli

Catching up: Your First Aesthetic Critique

Jordan Matuszewski

Sidebar: on effective homework submission

When submitting your homework, bear in mind that Blackboard only previews a select few file formats. With your submissions, please always make sure you include:

1. A representative image (JPG or PNG) that I can include in my lecture slides.
2. Any text included in your notebooks or code extracted in a format that I can easily read (TXT, DOCX, PDF)

- For instance, save your ipynb to PDF and upload the PDF as well

3. Upload image and text as separate files instead of hiding everything instead an archive format (such as ZIP or RAR)
4. If you do have to use an archive format, make sure it is universally readable (ie, use ZIP and not RAR or other more esoteric formats)

Altair interactive documents can be included into my slides by some fiddly work from an HTML file that includes the graph.

Data Abstraction - how do we represent data?

Types of Data

Munzner defines 5 fundamental data types:

1. Items
2. Attributes
3. Links
4. Positions
5. Grids

Items

An **item** is a discrete individual entity - such as a row in a tidy table, or a node in a network.

Attributes

Also called **variables** or **data dimensions** or **dimensions** (thought Munzner reserves *dimension* for the visual channels of spatial position - ie X/Y/Z-positioning)

Attributes

Categorical Data

Categorical (or **nominal**) data does not intrinsically support arithmetic operations or a total ordering - we can tell the difference between identities but not much more. Examples include names (of brands, people, places, …), movie genres, file types.

Attributes

Ordered Data

Data that has an implicit (total) ordering we call **Ordered Data**.
It subdivides into **ordinal** and **quantitative** data, depending on whether or not arithmetic operations are meaningful.
Examples include shirt size (ordinal), rankings (ordinal), height (quantitative), weight (quantitative), etc.

Attributes

Ordered - Quantitative Data

Quantitative data itself has several possibly relevant further subdivisions. We can distinguish between integers ($\mathbb{N}$) or reals ($\mathbb{R}$).
We can distinguish between **interval** data and **ratio** data - for interval data, the difference between values is meaningful; for ratio data, the ratio between values is meaningful. Interval data may not have a meaningful 0-value, while ratio data does.

Attributes

Ordered - Sequential / Diverging

Sequential data goes from a minimum value to a maximum value. It does not have to have a meaningful basepoint or 0-value.
Examples: height or weight of a person, course grades, taxation brackets.
Diverging ordinal data instead emerges from one *neutral* center in two (or, rarely, more) directions of progressively more extreme values.
Examples: temperature (above/below freezing), altitude (above/below sea level), 2-party election forecast (favor one or the other).

Attributes

Ordered - Cyclic

Cyclic “ordinal” data would not necessarily qualify as ordered in a mathematical sense - but features that arrange in a recurrent way impact visualization choices. An attribute is cyclic if its values wrap around to a starting point after a while.
Examples: time of day, day of week, day of year, year of century; but also compass directions, phase of periodic dynamics.

Links

Links are connections between items. Usually (in networks/graphs), a link will connect two items to each other - but there are interesting phenomena when allowing *hypergraphs* or *simplicial complexes* as data representation.
Examples: *friends*/*follows* on social networks, connectivity of roads or along train lines, the author/paper bipartite graph, family relationships, organizational charts.

Positions

Position data corresponds to locations in spatial data - primarily as 2-dimensional or 3-dimensional points.

Grids

Grid data in Munzner's usage is for spatial data, primarily as sampling strategies for continuously varying attributes - encoding both geometric and topological properties of the sampling domain.
Examples: square grid, triangular grid, hexagonal grid, adaptive grid, Voronoi cells.

Types of Datasets

Munzner identifies 4 basic types of datasets, each specified by some collection of the preceding data types:

Tables	**Networks & Trees**	**Fields**	**Geometry**
Items	Items (nodes)	Grids	Items
Attributes	Links	Positions	Positions
	Attributes (both for items and links)	Attributes	

Types of Datasets

Munzner identifies 4 basic types of datasets, each specified by some collection of the preceding data types:

Tables

This is the typical spread sheet data. Of particular interest is the notion of **tidy data** (Hadley Wickham):

1. Each variable must have its own column.
2. Each observation must have its own row.
3. Each value must have its own cell.

Tabular data also includes multidimensional tables (data hypercubes; tensors) that have composite keys.

Fields and Sampling Grids

The **field** dataset type concerns data that varies continuously and not discretely over some (geometric) domain.
We may distinguish between **scalar fields** that assign one attribute value to each point, and **vector fields** or even **tensor fields** that assign a vector (or matrix, or tensor) of values to each point.
Examples: Temperature, altitude, water depth, wind direction, local coordinate frames, stress tensors.
Core requirement for interacting with field data is to have access to a determined sampling grid that discretizes the field information.
A type of field data over a 1-dimensional domain is **time-varying data**. Not all data that contains time attributes are time-varying, but data that requires time attributes to completely specify **query keys** are what we mean by time-varying.

An Overview of the Grammar of Graphics

Six layers of a specification

Wilkinson breaks up the full specification of a statistical graphic into 6 separate specification steps (and Hadley Wickham breaks it up further by introducing the notion of **layers**)

1. **Data**: a set of operations that create **variables** from **datasets**.
2. **Trans**: variable transformations (eg `rank`).
3. **Scale**: scale transformations (eg `log`).
4. **Coord**: coordinate system (eg *polar*).
5. **Element**: graphical marks (*points*? *lines*?) and their aesthetic attributes (eg *color*)
6. **Guide**: guidance annotation components (*axes*, *legends*, etc)

Specifying a graphic

Wilkinson gives as an example the following specification and graphic:

Specifying a graphic

I found a data portal from the UN.

In R - calling the API and reorganizing the data:

- Code

```
# A tibble: 6 × 4
  country                          year    CBR   CDR
  <chr>                            <chr> <dbl> <dbl>
1 Algeria                          1990   30.8  5.73
2 Argentina                        1990   22.0  7.74
3 Bolivia (Plurinational State of) 1990   35.8 11.9
4 Brazil                           1990   24.8  7.16
5 Canada                           1990   15.5  6.95
6 Chile                            1990   22.2  5.62
```

Specifying a graphic

In R/ggplot2:

- Code

Specifying a graphic

```
ELEMENT: point(position(birth*death), size(0), label(country))
ELEMENT: contour(position(
           smooth.density.kernel.epanechnikov.joint(birth*death)),
           color.hue()
         )
GUIDE: form.line(position((0,0),(30,30)), label("Zero Population Growth"))
GUIDE: axis(dim(1), label("Birth Rate))
GUIDE: axis(dim(2), label("Death Rate))
```

```
ggplot(plot_data, aes(CBR, CDR)) +
  geom_density2d(aes(color=..level..)) +
  geom_text(aes(label=country), size=2) +
  geom_abline() +
  annotate(geom="text", x=20, y=22, angle=50, size=3, label="Zero population growth") +
  scale_x_continuous(limits=c(0,60)) +
  scale_y_continuous(limits=c(0,30)) +
  scale_color_distiller(palette="Spectral", guide="none") +
  xlab("Birth Rate") + ylab("Death Rate")
```

Specifying a graphic

I found a data portal from the UN.

In Python - calling the API and compiling a query URL:

- Code

indicator	location	Crude birth rate	Crude death rate
0	Algeria	30.762	5.731
1	Argentina	21.989	7.743
2	Bolivia (Plurinational State of)	35.840	11.925
3	Brazil	24.844	7.158
4	Canada	15.458	6.954
5	Chile	22.190	5.625
6	Costa Rica	26.734	3.758
7	Ecuador	29.969	5.678
8	Ethiopia	50.052	20.055
9	France	13.338	9.334
10	Gambia	45.733	16.102
11	Germany	11.297	11.625
12	Guinea	46.318	19.494
13	Haiti	38.263	14.029
14	Hungary	12.091	14.033
15	Iraq	39.409	10.565
16	Italy	10.009	9.568
17	Jamaica	25.322	6.861
18	Libya	29.479	4.663
19	Malaysia	27.894	4.656

Specifying a graphic

In Python / altair:

- Code

Note that contour density plots are still a pending feature for Vega-Lite, and therefore still a pending feature for Altair. We could probably reproduce Wilkinson's plot completely, but would have to compute the contour curves ourselves at considerable effort.

Specifying a graphic

```
ELEMENT: point(position(birth*death), size(0), label(country))
ELEMENT: contour(position(
           smooth.density.kernel.epanechnikov.joint(birth*death)),
           color.hue()
         )
GUIDE: form.line(position((0,0),(30,30)), label("Zero Population Growth"))
GUIDE: axis(dim(1), label("Birth Rate))
GUIDE: axis(dim(2), label("Death Rate))
```

```
altair.Chart(data).mark_text(fontSize=10).encode(
  x="Crude birth rate:Q",
  y="Crude death rate:Q",
  text="location:N") + altair.Chart(
  pandas.DataFrame({"x": [0,30],"y":[0,30]})
).mark_line().encode(
  x="x:Q", y="y:Q"
) + altair.Chart(
  pandas.DataFrame({"x": [23], "y": [25],
   "text": "Zero Population Growth"})
).mark_text(angle=305, fontSize=12).encode(
  x="x:Q", y="y:Q", text="text:N"
)
```