Introduction to ggplot2

Dawn Koffman
Office of Population Research
Princeton University
November 2021
Part 1: Concepts and Terminology

“It’s hard to succinctly describe how ggplot2 works because it embodies a deep philosophy of visualisation.”

- From https://ggplot2.tidyverse.org
R Package: ggplot2

Used to produce statistical graphics, author = Hadley Wickham

"attempt to take the good things about base and lattice graphics and improve on them with a strong, underlying model “


based on *The Grammar of Graphics* by Leland Wilkinson, 2005

"... describes *the meaning* of what we do when we construct statistical graphics ... More than a taxonomy ... Computational system based on the underlying mathematics of representing statistical functions of data."

- does not limit developer to a set of pre-specified graphics

adds some concepts to grammar which allow it to work well with R
ggplot2 provides two ways to produce plot objects:

**qplot()**  # *quick plot* – not covered in this workshop

uses some concepts of *The Grammar of Graphics*, but doesn’t provide full capability and
designed to be very similar to plot() and simple to use

may make it easy to produce basic graphs

but

may delay understanding philosophy of ggplot2

---

**ggplot()**  # *grammar of graphics plot* – focus of this workshop

provides fuller implementation of *The Grammar of Graphics*

may have steeper learning curve but allows much more flexibility when building graphs
Grammar Defines Components of Graphics

data: in ggplot2, data must be stored as an R data frame

coordinate system: describes 2-D space that data is projected onto
  - for example, Cartesian coordinates, polar coordinates, map projections, ...

geoms: describe type of geometric objects that represent data
  - for example, points, lines, polygons, ...

aesthetics: describe visual characteristics that represent data
  - for example, position, size, color, shape, transparency, fill

scales: for each aesthetic, describe how visual characteristic is converted to display values
  - for example, log scales, color scales, size scales, shape scales, ...

stats: describe statistical transformations that typically summarize data
  - for example, counts, means, medians, regression lines, ...

facets: describe how data is split into subsets and displayed as multiple small graphs
Workshop Data Frame

extract from 2012 World Population Data Sheet produced by Population Reference Bureau

includes data for 158 countries where mid-2012 population >= 1 million

variables:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>country</td>
<td>country name</td>
</tr>
<tr>
<td>pop2012</td>
<td>population mid-2012 (millions)</td>
</tr>
<tr>
<td>imr</td>
<td>infant mortality rate*</td>
</tr>
<tr>
<td>tfr</td>
<td>total fertility rate*</td>
</tr>
<tr>
<td>le</td>
<td>life expectancy at birth</td>
</tr>
<tr>
<td>leM</td>
<td>male life expectancy at birth</td>
</tr>
<tr>
<td>leF</td>
<td>female life expectancy at birth</td>
</tr>
<tr>
<td>area</td>
<td>(Africa, Americas, Asia &amp; Oceania, Europe)</td>
</tr>
<tr>
<td>region</td>
<td>(Northern Africa, Western Africa, Eastern Africa, Middle Africa, North America, Central America, Caribbean, South America, Western Asia, South Central Asia, Southeast Asia, East Asia, Oceania, Northern Europe, Western Europe, Eastern Europe, Southern Europe)</td>
</tr>
</tbody>
</table>

*definitions:
infant mortality rate – annual number of deaths of infants under age 1 per 1,000 live births

total fertility rate – average number of children a woman would have assuming that current age-specific birth rates remain constant throughout her childbearing years
Create a Plot Object

creates a plot object that can be assigned to a variable

can specify data frame and aesthetic mappings (visual characteristics that represent data)

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr))
```

country | pop2012 | tfr | le | area  
--- | --- | --- | --- | ---  
Algeria | 37.4 | 2.9 | 73 | Africa  
Egypt | 82.3 | 2.9 | 72 | Africa  
Libya | 6.5 | 2.6 | 75 | Africa  
Morocco | 32.6 | 2.3 | 72 | Africa  
South Sudan | 9.4 | 5.4 | 52 | Africa  
Sudan | 33.5 | 4.2 | 60 | Africa  
Tunisia | 10.8 | 2.1 | 75 | Africa  
Benin | 9.4 | 5.4 | 56 | Africa  
Burkina Faso | 17.5 | 6.0 | 55 | Africa  
Cote d'Ivoire | 20.6 | 4.6 | 55 | Africa  
Gambia | 1.8 | 4.9 | 58 | Africa  
Ghana | 25.5 | 4.2 | 64 | Africa  

x-axis position indicates le value  
y-axis position indicates tfr value
Adding a Layer

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr, color=area))
p + geom_point(size=4)
```
Layer

purpose:

display the data –
allows viewer to see patterns, overall structure, local structure, outliers, ...

display statistical summaries of the data –
allows viewer to see counts, means, medians, IQRs, model predictions, ...

data and aesthetics (mappings) may be inherited from ggplot() object or added, changed, or dropped within individual layers

most layers contain a geom ... the fundamental building block of ggplot2
full specification: geom_xxx(mapping, data, stat, position, ...)

each geom_xxx() has a default stat (statistical transformation) associated with it, but the default statistical transformation may be changed using stat parameter
Adding a *geom* Layer

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",")
p <- ggplot(data=w, aes(x=le, y=tfr, color=area))

p + geom_blank()
p + geom_point()
p + geom_jitter()
p + geom_count()
```
Adding a `geom` Layer: Connect Points

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\n"")
p <- ggplot(data=w, aes(x=le, y=tfr, color=area))
p + geom_line()
p + geom_step()
p + geom_path()
```
Displaying Data and Statistical Summary

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr))

p + geom_point(shape=1) + geom_smooth()
```

```r
p + geom_point(shape=1) + geom_smooth(method="lm", se=FALSE)
```
Displaying Statistical Summary

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","")
p <- ggplot(data=w, aes(x=area))
p + geom_bar()
```
Already Transformed Data

```r
wb <- read.csv(file="WDS2012areabins.csv", head=TRUE, sep="",""

wb

<table>
<thead>
<tr>
<th>bin</th>
<th>area</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Africa</td>
<td>48</td>
</tr>
<tr>
<td>2</td>
<td>Americas</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>Asia/Oceania</td>
<td>49</td>
</tr>
<tr>
<td>4</td>
<td>Europe</td>
<td>36</td>
</tr>
</tbody>
</table>
```

```r
p <- ggplot(data=wb, aes(x=area, y=count))
p + geom_col()
# OR
p + geom_bar(stat="identity")
```

**geom_bar**: height of bar proportional to number of observations in each group.

**geom_col**: leaves data as is.

*geom_bar* uses *count* stat by default. *geom_col* uses *identity* stat.
Displaying Distributions

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le))

p + geom_histogram()

p + geom_freqpoly()

p + geom_freqpoly(aes(color=area))
```
Displaying Statistical Summaries

w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","n")
p <- ggplot(data=w, aes(x=area, y=le))

p + geom_boxplot()
p + geom_violin()
For more detail, see Rstudio’s Cheat Sheet, Data Visualization with ggplot2, at https://www.rstudio.com/resources/cheatsheets/
two variables, both continuous

- geom_label
- geom_jitter
- geom_point
- geom_quantile
- geom_rug
- geom_smooth
- geom_text

two variables, discrete x, continuous y

- geom_col
- geom_boxplot
- geom_dotplot
- geom_violin

two variables, discrete x, discrete y

- geom_count

For more detail, see Rstudio's Cheat Sheet, Data Visualization with ggplot2, at [https://www.rstudio.com/resources/cheatsheets/](https://www.rstudio.com/resources/cheatsheets/)
two variables, visualizing error

- geom_crossbar
- geom_errorbar
- geom_errorbarh
- geom_linerange
- geom_pointrange

two variables, continuous bivariate distribution

- geom_bin2d
- geom_density2d
- geom_hex

two variables, continuous function

- geom_area
- geom_line
- geom_step

two variables, maps

- geom_map

Full specification of each geom at: 
http://ggplot2.tidyverse.org/reference/#section-layer-geoms

For more detail, see Rstudio’s Cheat Sheet, Data Visualization with ggplot2, at https://www.rstudio.com/resources/cheatsheets/
Aesthetics

describe visual characteristics that represent data
  - for example, x position, y position, size, color (outside), fill (inside),
    point shape, line type, transparency

each layer inherits default aesthetics from plot object
  - within each layer, aesthetics may added, overwritten, or removed

most layers have some required aesthetics and some optional aesthetics

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr, color=area))
p + geom_point() + geom_smooth(method="lm", se=FALSE)
```
Add or Remove Aesthetic Mapping

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\n")
p <- ggplot(data=w, aes(x=le, y=tfr, color=area))

add aesthetic mapping
p + geom_point(aes(shape=area)) + geom_smooth(method="lm", se=FALSE)

remove aesthetic mapping
p + geom_point(aes(color=NULL)) + geom_smooth(method="lm", se=FALSE)
```
Aesthetic Mapping vs. Parameter Setting

aesthetic mapping
  data value determines visual characteristic
  use `aes()`

setting
  constant value determines visual characteristic
  use layer parameter

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\n)p <- ggplot(data=w, aes(x=le, y=tfr))
p + geom_point(aes(color=area))
P + geom_point(color="red")
```
```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",")
w$tfrGT2 <- w$tfr > 2
p <- ggplot(data=w, aes(x=area, fill=tfrGT2))
p + geom_bar()
p + geom_bar(position="stack")
p + geom_bar(position="dodge")
p + geom_bar(position="fill")
```
Bar Width

w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","

p <- ggplot(data=w, aes(x=area))

p + geom_bar()

p + geom_bar(width=.9) # default

p + geom_bar(width=.5)

p + geom_bar(width=.97)
Position

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr))

p + geom_point()
p + geom_point(position="jitter")
p + geom_jitter()

For reproducible jittering, set a seed ... new “seed” argument for position_jitter(), as of 3.1.0:
p + geom_point(position=position_jitter(seed=1))
```
Transparency

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr))

p + geom_point(size=3, alpha=1/2)

p + geom_jitter(size=4, alpha=1/2)


techniques for overplotting: adjusting symbol size, shape, jitter, and transparency
```
Coordinate System

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",")
p <- ggplot(w, aes(x=factor(1), fill=area))

p + geom_bar()

p + geom_bar() + coord_flip()

p + geom_bar() + coord_polar(theta="y")

p + geom_bar() + coord_polar(theta="y", direction=-1)
```
Data Frame

each plot layer may contain data from a different data frame

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",")
africa <- subset(w, area=="Africa")
europe <- subset(w, area=="Europe")

p <- ggplot(data=europe, aes(x=le, y=tfr))
p + geom_jitter(color="blue") +
  geom_jitter(data=africa, color="red")

africa_europe <- rbind(africa, europe)
p <- ggplot(data=africa_europe, aes(x=le, y=tfr, color=area))
p + geom_jitter()

OR

p <- ggplot(data=rbind(africa, europe), aes(le, y=tfr, color=area))
p + geom_jitter()
```
Labels

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\nwna <- subset(w, region=="Northern Africa")
p <- ggplot(data=wna, aes(x=le, y=tfr))
```

```
p + geom_point() + geom_text(aes(label=country), nudge_y=.2, size=4) + xlim(50,80)
```

```
p + geom_point() + geom_label(aes(label=country), nudge_y=.3, size=3) + xlim(50,80) + ylim(2,6)
```
Labels

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\n"
labelset <- c("South Sudan", "Sudan", "Libya", "Tunisia")

p <- ggplot(data=subset(w, region=="Northern Africa"),
  aes(x=le, y=tfr))
p +
  geom_point() +
  geom_text(data=subset(w, country %in% labelset),
    aes(label=country), nudge_y = .2, color="blue") +
  xlim(50,80)
```

![Graph showing population data for selected countries in Northern Africa]
Non-Overlapping Labels

install.packages("ggrepel")
library("ggrepel")
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""")
wna <- subset(w, region=="Northern Africa")
p <- ggplot(data=wna, aes(x=le, y=tfr))

p + geom_point() + geom_text_repel(aes(label=country), size=4) + xlim(50, 80)

p + geom_point() + geom_label_repel(aes(label=country), size=4) + xlim(50, 80) + ylim(2, 6)
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",")
wna <- subset(w, region=="Northern Africa")
p <- ggplot(data=wna, aes(x=le, y=tfr))

p + geom_point() +
annotate("text", x=55, y=5.5, label="South Sudan", color="red") +
annotate("text", x=62, y=4.3, label="Sudan", color="red") +
annotate("rect", xmin = 71.5, xmax = 75.5, ymin = 1.9, ymax = 3.1,
alpha = .2) +
ggtitle("Northern Africa") + xlab("life expectancy")
Scale

controls the mapping from data to aesthetic
“takes data and turns it into something that can be perceived visually”
color and fill, shape, size, position

acts as a function from the data space to a place in the aesthetic space

provides axes or legends (“guides”) to allow viewer to perform inverse mapping from aesthetic space back to data space

required for every aesthetic ... so ggplot2 always provides a default scale

```
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

w$tfrGT2 <- w$tfr > 2

p <- ggplot(data=w, aes(x=area, fill=tfrGT2))
p + geom_bar(color="black")
```

equivalent to
```
p + geom_bar(color="black") +
scale_fill_discrete()
```
equivalent to
```
p + geom_bar(color="black") +
scale_fill_hue()
```

colors equally spaced around color wheel
Fill Scales

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""
w$tfrGT2 <- w$tfr > 2
p <- ggplot(data=w, aes(x=area, fill=tfrGT2))

p + geom_bar(color="black") + scale_fill_grey()

p + geom_bar(color="black") + scale_fill_brewer()
```
library(RColorBrewer)
display.brewer.all()

w <- read.csv(file="WDS2012.csv", 
head=TRUE, sep="","")
w$tfrGT2 <- w$tfr > 2

p <- ggplot(data=w, 
aes(x=area, fill=tfrGT2))

p + geom_bar(color="black") + 
scale_fill_brewer(palette="Accent")
Manual Scales

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

w$tfrGT2 <- w$tfr > 2
p <- ggplot(data=w, aes(x=area, fill=tfrGT2))

p + geom_bar(color="black") +
  scale_fill_manual(values=c("red","blue"),
                    labels=c("no", "yes"))

p + geom_point(aes(x=le, y=tfr,
                    shape=area, fill=NULL), size = 3) +
  xlab("life expectancy") +
  scale_shape_manual(values=c(1,16,2,8))
```

typical scale arguments: values
labels
breaks
limits
name
Position Scales

w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""")

p <- ggplot(data=w, aes(x=le, y=tfr))
p + geom_jitter()

p + geom_jitter() + scale_y_reverse()

p <- ggplot(data=w, aes(x=le, y=pop2012))
p + geom_jitter()

p + geom_jitter() + scale_y_log10(breaks=c(10, 100, 1000), labels=c(10,100,1000))
Theme

controls appearance of **non-data elements**
... does not affect how data is displayed by `geom_xxx()` function

helps make plot visually pleasing by allowing addition/modification/deletion of titles, axis labels, tick marks, axis tick labels and legends

theme elements **inherit** properties from other theme elements, for example:
Theme: Titles, Tick Marks, and Tick Labels

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr))
p + geom_jitter() + ggtitle("Life Expectancy and TFR") +
xlab("life expectancy (years)") +
ylab("total fertility rate (tfr)") +
scale_x_continuous(breaks=seq(50,80,by=5),
    labels=c(50,"fifty-five",60,65,70,75,80)) +
theme(plot.title=element_text(color="blue", size=24, hjust= 0.5),
    axis.title=element_text(size=14,face="bold"),
    axis.title.x=element_text(color="green"),
    axis.text=element_text(size=14),
    axis.text.y=element_text(color="black"),
    axis.text.x=element_text(color="purple", size=20),
    axis.ticks.y=element_blank())
```

Life Expectancy and TFR

![Graph showing Life Expectancy and TFR](image-url)
Theme: Legends

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

w$tfrGT2 <- w$tfr > 2

p <- ggplot(data=w, aes(x=area, fill=tfrGT2))

p + geom_bar() +
    scale_fill_manual(name="TFR value",
                      values = c("red","blue"),
                      labels=c("<=2", ">2")) +
    theme(legend.position="left",
          legend.text.align=1)

p + geom_point(aes(x=le, y=tfr,
                      shape=area, fill=NULL), size = 3) +
    xlab("life expectancy") +
    scale_shape_manual(name="Area: ",
                       values=c(1,16,2,8)) +
    theme(legend.key=element_blank(),
          legend.direction="horizontal",
          legend.position="bottom")
```
Theme: Overall Look

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","

p <- ggplot(data=w, aes(x=le, y=tfr))

p + geom_point() + theme_gray()

p + geom_point() + theme_bw()

p + geom_point() + theme_classic()

p + geom_point() + theme_minimal()

to change default theme use theme_set() ... for example, theme_set(theme_classic())
```
Themes - More Overall Looks

```
install.packages("ggthemes")
library("ggthemes")
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""
p <- ggplot(data=w, aes(x=le, y=tfr, color=area))
```

```
p + geom_point() + theme_base()
```

```
p + geom_point() + theme_excel() + scale_color_excel()
```

```
p + geom_point() + theme_wsj()
```

```
p + geom_point() + theme_igray()
```
Themes - More Overall Looks

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr, color=area))

p + geom_point() + theme_stata() + scale_color_stata()
```

![Graph with theme_stata() and theme_economist() with scale_y_continuous(pos="right")](image)

```r
p + geom_point() + theme_economist() + scale_color_economist() + scale_y_continuous(pos="right")
```
Themes - More Overall Looks

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(data=w, aes(x=le, y=tfr, color=area))

p + geom_point() +
theme_fivethirtyeight() +
scale_color_fivethirtyeight
```

```r
p + geom_point() +
theme_solarized(light=FALSE) +
scale_color_manual(values=c("red", "green", "yellow", "purple"))
```
Facets

split data into subsets and plot each subset on a different panel
- show data as "small multiples"

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\n)p <- ggplot(data=w, aes(x=le, y=imr)) + geom_jitter()
p + facet_grid(. ~ area)
p + facet_grid(area ~ .)
```
```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""
w$tfrGT2 <- w$tfr > 2
p <- ggplot(data=w, aes(x=le, y=imr)) + geom_jitter()

p + facet_grid(area ~ tfrGT2, labeller="label_both")
p + facet_grid(tfrGT2 ~ area, labeller="label_both", margins=TRUE)
```
Saving Graphs

\texttt{ggsave()}

- saves last plot displayed
- requires file name to be supplied
- uses file name extension to determine file type:
  \texttt{.ps \ .eps \ .tex \ .pdf \ .jpg \ .tiff \ .png \ .bmp \ .svg \ .wmf} (windows only)
- uses size of current graphics device for default size

\begin{verbatim}
  w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","")
ggplot(data=w, aes(x=le, y=tfr, color=area)) + geom_point()

ggsave(file="le_tfr1.jpg")
ggsave(file="le_tfr2.jpg", scale=2)
ggsave(file="le_tfr3.jpg", width=5, height=5, unit="in")

ggsave(file="le_tfr4.png")
ggsave(file="le_tfr5.pdf")
\end{verbatim}
Part 2: Examples
Contents and Purpose of ggplot2 Graphs

ggplot2 graph is typically created to show:

- data
- data + annotation
- statistical summary
- statistical summary + annotation
- data + statistical summary
- data + statistical summary + annotation

purpose of graph:

- **explore** data to increase understanding of data

- **communicate** about data ... often by showing data and/or statistical summary **plus** annotation


Show Data

w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""
popLT300 <- subset(w,pop2012<300)

p <- ggplot(data=popLT300,
aes(x=area, y=tfr, size=pop2012))
p + geom_jitter(position=position_jitter(w=.2, h=.1),shape=21)
scale_size_area(max_size=10)

Why is it important to show raw data?
Anscombe’s Quartet

4 data sets that have nearly identical summary statistics

each has 11 non-missing pairs of values

constructed in 1973 by statistician Francis Anscombe to demonstrate importance of graphing data and effect of outliers

<table>
<thead>
<tr>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 3</th>
<th>Set 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>y</td>
<td>x</td>
<td>y</td>
</tr>
<tr>
<td>10.0</td>
<td>8.04</td>
<td>10.0</td>
<td>9.14</td>
</tr>
<tr>
<td>8.0</td>
<td>6.95</td>
<td>8.0</td>
<td>8.14</td>
</tr>
<tr>
<td>13.0</td>
<td>7.58</td>
<td>13.0</td>
<td>8.74</td>
</tr>
<tr>
<td>9.0</td>
<td>8.81</td>
<td>9.0</td>
<td>8.77</td>
</tr>
<tr>
<td>11.0</td>
<td>8.33</td>
<td>11.0</td>
<td>9.26</td>
</tr>
<tr>
<td>14.0</td>
<td>9.96</td>
<td>14.0</td>
<td>8.10</td>
</tr>
<tr>
<td>6.0</td>
<td>7.24</td>
<td>6.0</td>
<td>6.13</td>
</tr>
<tr>
<td>4.0</td>
<td>4.26</td>
<td>4.0</td>
<td>3.10</td>
</tr>
<tr>
<td>12.0</td>
<td>10.84</td>
<td>12.0</td>
<td>9.13</td>
</tr>
<tr>
<td>7.0</td>
<td>4.82</td>
<td>7.0</td>
<td>7.26</td>
</tr>
<tr>
<td>5.0</td>
<td>5.68</td>
<td>5.0</td>
<td>4.74</td>
</tr>
</tbody>
</table>

SUMMARY STATISTICS

<table>
<thead>
<tr>
<th></th>
<th>Set 1</th>
<th>Set 2</th>
<th>Set 3</th>
<th>Set 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean value of x</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>mean value of y</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
</tr>
<tr>
<td>variance of x</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>variance of y</td>
<td>4.1</td>
<td>4.1</td>
<td>4.1</td>
<td>4.1</td>
</tr>
<tr>
<td>correlation between x and y</td>
<td>0.816</td>
<td>0.816</td>
<td>0.816</td>
<td>0.816</td>
</tr>
<tr>
<td>linear regression (best fit) line is:</td>
<td>y=0.5x+3</td>
<td>y=0.5x+3</td>
<td>y=0.5x+3</td>
<td>y=0.5x+3</td>
</tr>
</tbody>
</table>

Data + Annotation

```r
p <- ggplot(data=popLT300,
    aes(x=area, y=tfr, size=pop2012))
p + geom_jitter(position=
    position_jitter(w=.2, h=.1),shape=21) +
scale_y_continuous(breaks=
    c(1,2,3,4,5,6,7)) +
scale_size_area(max_size=10) +
annotate("text", x=1.3,y=7.1,
    label="Niger", size=4) +
labs(title="Country Total Fertility Rate
(TFRs), 2012",
    x="\nNote: United States, China and
    India are not included.\n",
y="Total\nFertility\nRate\n(TFR)",
    size="2012 Population\n(millions)") +
theme_bw() +
theme(axis.title.x=element_text(size=10,
    hjust=0),
    axis.title.y=element_text(angle=0)
    legend.key=element_blank(),
    legend.text.align=1)
```
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","")

p <- ggplot(data=w, aes(x=area, y=tfr, size=pop2012))
p + geom_jitter(position=
    position_jitter(w=.2, h=.1),
    shape=21, fill="gray") +
scale_y_continuous(breaks=
c(1,2,3,4,5,6,7)) +
scale_size_area(breaks=
c(50,100,200,300,1000),
    max_size=18) +
theme_bw() +
theme(axis.title.x=element_blank(),
    axis.title.y=element_text(angle=0)
    legend.key=element_blank(),
    legend.text.align=1)
Data + Statistical Summary

```r
w <- read.csv(file="WDS2012.csv", 
               head=TRUE, sep="","\n")

p <- ggplot(w, aes(x=area,
                   y=tfr,color=area))

p + geom_boxplot(outlier.size=0) +
  geom_jitter(position=
               position_jitter(h=.1))
```
Data + Statistical Summary + Annotation

```r
p <- ggplot(data=subset(w, area=="Africa"),
aes(x=reorder(factor(region), tfr,FUN=median),
y=tfr, color=region))
p + geom_boxplot(outlier.size=0) +
  geom_jitter(position=
    position_jitter(w=.2, h=0)) +
  annotate("text", x=1.2, y=5.5,
    label="South Sudan", size=4) +
  annotate("text", x=3.3, y=1.5,
    label="Mauritius", size=4) +
  annotate("text", x=4.8, y=7.1,
    label="Niger", size=4) +
  annotate("text", x=4, y=3.2,
    label="Gabon", size=4) +
labs(title="Country TFR's for Africa, 2012",
  x="", y="TFR") +
theme(axis.ticks.x=element_blank(),
  axis.title.y=element_text(angle=0),
  legend.position="none")
```
Statistical Summary

violin plot:

kernel density estimates, mirrored to have a symmetrical shape

allows visual comparison of data distribution of several groups

```
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","")
p <- ggplot(w, aes(x=area, y=tfr, color=area))
p + geom_violin()
```
Statistical Summaries

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(w, aes(x=reorder(factor(area),tfr,FUN="median"), y=tfr)) +
  geom_violin() + geom_boxplot(width=.1, outlier.size=0) +
  scale_y_continuous(breaks=c(1,2,3,4,5,6,7)) +
  theme(axis.title.y=element_text(angle=0,size=12),
        axis.text.y=element_text(color="black", size=12),
        axis.text.x=element_text(color="black", size=12),
        legend.position="none") +
  labs(title="Country TFRs: Density Distribution, Median and IQR by Area, 2012",
       x="", y="TFR")
```

Country TFRs: Density Distribution, Median and IQR by Area, 2012

![Graph showing TFRs by area with density distribution, median, and IQR]
Statistical Summary

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- ggplot(w, aes(x=le, color=area))
p + geom_line(stat="density", size=1.5)

p <- ggplot(w, aes(x=le, fill=area))
p + geom_density(alpha=.4)
```

density distribution
Statistical Summary + Annotation

w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","

p <- ggplot(w, aes(x=le, fill=area))

p + geom_density(alpha=.4) +
  scale_fill_manual(values=c("red", "green", "blue", "yellow")) +
  scale_x_continuous(breaks=c(45,50,55,60,65,70,75,80,85)) +
  theme(axis.text=element_text(color="black", size=12)) +
  labs(title="Distribution of Life Expectancy, by Area, 2012", x="life expectancy")

Distribution of Life Expectancy, by Area, 2012
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","")
p <- ggplot(w, aes(x=le))
p + geom_histogram(fill="darkgray", binwidth=1)

p + geom_freqpoly(color="red", size=1, binwidth=1)

p + geom_histogram(fill="darkgray", binwidth=1)

p + geom_freqpoly(color="red", size=1, binwidth=1)
Show Data

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="", )
p <- ggplot(data=subset(w, area=="Africa"), aes(x=tfr, y=reorder(factor(country), tfr)))
p + geom_point()
```
Show Data

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""

p <- gplot(data=subset(w, area=="Africa"),
aes(x=tfr,y=reorder(factor(country), tfr)))

p + geom_segment(aes(yend=country,xend=0)) +
geom_point() +
theme_minimal() +
scale_x_continuous(breaks =
c(0,1,2,3,4,5,6,7)) +
labs(x="Total Fertility Rate (TFR), y="",
title="Total Fertility Rates in Africa, by Country, 2012") +
theme(panel.grid.major.y=element_blank(),
axis.ticks=element_blank())
```
```r
w <- read.csv(file="WDS2012.csv",
               head=TRUE, sep="","
)

p <- ggplot(data=subset(w, area=="Africa"),
            aes(x=tfr, y=reorder(factor(country),tfr)))

p + geom_text(aes(x=tfr-.1, label=country,
                  hjust=1), size=4) +
geom_point() +
theme_minimal() +
scale_x_continuous(breaks=c(1,2,3,4,5,6,7),
                   limits=c(0,8)) +
labs(x="", y="",
     title="Total Fertility Rates (TFRs) in
           Africa, by Country, 2012") +
theme(panel.grid.major.y=element_blank(),
      axis.text.y=element_blank(),
      axis.ticks=element_blank())
```

Show Data

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="", )
a <- subset(w, area=="Africa")
a$region <- factor(a$region, levels = c("Northern Africa","Southern Africa", "Western Africa", "Middle Africa", "Eastern Africa" ))
p <- ggplot(data=a,aes(x=tfr,
    y=reorder(factor(country),tfr)))
p + geom_segment(aes(yend=country,xend=0)) + geom_point() + scale_x_continuous(breaks=
    c(0,1,2,3,4,5,6,7)) +
labs(x="Total Fertility Rate (TFR)", y="", title="Total Fertility Rates (TFRs) in Africa, by Country, 2012") +
theme(
    axis.text=element_text(color="black"),
    strip.text.y=element_text(size=9),
    strip.background=element_rect(fill="white"),
    panel.grid.major.y=element_blank(),
    panel.grid.minor.x=element_blank(),
    axis.ticks=element_blank()) +
    facet_grid(region ~ .)
```
Show Data

```r
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="",""
)a <- subset(w,area=="Africa")
a$region <- factor(a$region,levels=c("Northern Africa","Southern Africa","Western Africa","Middle Africa","Eastern Africa" ))
p <- ggplot(data=a,aes(x=tfr,
        y=reorder(factor(country),tfr)))
p + geom_segment(aes(yend=country,xend=0)) + geom_point() + scale_x_continuous(breaks=c(0,1,2,3,4,5,6,7)) + labs(x="Total Fertility Rate (TFR)",
y="",title="Total Fertility Rates (TFRs) in Africa, by Country, 2012") + theme(axis.text=element_text(color="black"),
        strip.text.y=element_text(size=9),
        strip.background=element_rect(fill="white "),
        panel.grid.major.y=element_blank(),
        panel.grid.minor.x=element_blank(),
        axis.ticks=element_blank()) + facet_grid(region ~ ., scales="free_y")
```
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","")
a <- subset(w, area=="Africa")
a$region <- factor(a$region, levels=c("Northern Africa", "Southern Africa", "Western Africa", "Middle Africa", "Eastern Africa" ))
p <- ggplot(data=a, aes(x=tfr, y=reorder(factor(country), tfr)))
p + geom_segment(aes(yend=country, xend=0)) + geom_point() + scale_x_continuous(breaks=c(0,1,2,3,4,5,6,7)) + labs(x="Total Fertility Rate (TFR)", y="", title="Total Fertility Rates (TFRs) in Africa, by Country, 2012") + theme(axis.text=element_text(color="black"), strip.text.y=element_text(size=9), strip.background=element_rect(fill="white"), panel.grid.major.y=element_blank(), panel.grid.minor.x=element_blank(), axis.ticks=element_blank()) + facet_grid(region ~ ., scales="free_y", space="free_y")
w <- read.csv(file="WDS2012.csv", head=TRUE, sep="","\n")

w$tfrGT2 <- w$tfr > 2
p <- ggplot(data=w,
            aes(x=area, fill=tfrGT2))
p + geom_bar() + scale_fill_manual(name="TFR value",
values = c("red","blue"),
labels=c("<=2", ">2")) +
    theme(legend.text.align=1)

w$imrGT15 <- w$imr > 15
p <- ggplot(data=w,
            aes(x=area, fill=imrGT15))
p + geom_bar() + scale_fill_manual(name="IMR value",
values = c("red","blue"),
labels=c("<=15", ">15")) +
    theme(legend.text.align=1)
w <- read.csv(file="WDS2012.csv",
               head=TRUE, sep="","
)

p <- ggplot(data=w, aes(x=imr, y=tfr))
p + geom_point(aes(color=area)) +
scale_color_manual(values=
  c("red", "blue", "green", "orange")) +
scale_y_continuous(breaks=c(0,1,2,3,4,5,6,7),
  limits=c(0,7.8)) +
scale_x_continuous(breaks=
  c(0,15,30,45,60,75,90,105,120)) +
theme_bw() +
theme(legend.position="bottom",
      legend.direction="horizontal",
      legend.key=element_blank()) +
geom_vline(xintercept=15, linetype="dashed") +
geom_hline(yintercept=2, linetype="dashed") +
geom_smooth(method="lm", color="black", size=.8)
geom_rug(position="jitter", size=.1)
Part 3: Recap and Additional Resources
Recap

<table>
<thead>
<tr>
<th>country</th>
<th>tfr</th>
<th>imr</th>
<th>area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algeria</td>
<td>2.9</td>
<td>24</td>
<td>Africa</td>
</tr>
<tr>
<td>Egypt</td>
<td>2.9</td>
<td>24</td>
<td>Africa</td>
</tr>
<tr>
<td>Canada</td>
<td>1.7</td>
<td>5.1</td>
<td>Americas</td>
</tr>
<tr>
<td>United States</td>
<td>1.9</td>
<td>6.0</td>
<td>Americas</td>
</tr>
<tr>
<td>Armenia</td>
<td>1.7</td>
<td>11</td>
<td>Asia/Oceania</td>
</tr>
<tr>
<td>Azerbaijan</td>
<td>2.3</td>
<td>11</td>
<td>Asia/Oceania</td>
</tr>
<tr>
<td>Denmark</td>
<td>1.8</td>
<td>3.5</td>
<td>Europe</td>
</tr>
<tr>
<td>Estonia</td>
<td>2.5</td>
<td>3.3</td>
<td>Europe</td>
</tr>
</tbody>
</table>

construct graphs by considering:

- coordinate system
- which values will be represented by various visual characteristics (aesthetics)
- how values will mapped to visual characteristics (scales)
- geometric rendering (geom)
- whether data might be displayed as “small multiples” (facets)
- adding additional annotation
Additional ggplot2 Resources

official "Package ggplot2" documentation and help
- http://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf
- http://ggplot2.tidyverse.org

online ggplot2 user community
- http://groups.google.com/group/ggplot2
- http://stackoverflow.com/tags/ggplot2
- https://rstudio.com/resources/cheatsheets/

books
- R for Data Science (Chapter 3), by Hadley Wickham & Garrett Grolemund, online at https://r4ds.had.co.nz/data-visualisation.html, 2017.
Thank You!

Workshop Survey

https://docs.google.com/forms/d/e/1FAIpQLSfBLpW9fSVEPiTR9MCPrw4hSKX2eCrFr1Ri0hiqDokR8qXNg/viewform

Computing Training: https://researchcomputing.princeton.edu/workshops

Help Sessions: Tuesdays 10:30-11:30am and Thursdays 2:00-3:00pm
  https://researchcomputing.princeton.edu/education/help-sessions

Instructor email: dkoffman@princeton.edu