Introduction

This guide reviews commonly used features when producing charts in R. We will cover some of the material in the workshop, so you should refer to this guide for more detail. It should be a helpful introduction to graphing in R.

Example Data

```r
X<-c(1:10)  # example X values
Y<-c(1:10)  # example Y Values
XY<-data.frame(X,Y)  # Create an example data frame containing X and Y
```

Graphing Your Data

```r
ggplot(XY, aes(X, Y))  # Graph data
  # ggplot(data frame, aes(X values, Y values))
  # If you are not graphing from a data frame, set the first input to NULL:
  # ggplot(NULL, aes(X, Y))
```

Bar Chart

```r
+geom_bar(stat = "identity", width = 1, fill = "grey", color = "black")  # Bar Chart
```

Histogram

```
# geom_bar(stat=height calculation, width=bar width as a proportion,
# fill='bar color', color='bar outline color')

# stat height calculation:

# 1) 'identity' plots the value associated with each X category (use when
# graphing from a frequency table)

# 2) 'count' plots the number of cases at each x position (use when graphing
# observational data)
```

---

1 Written by Anna Solow
Frequency Polygons

`+geom_freqpoly(stat = "identity")`  # Histogram as a line chart

# geomfreqpoly(stat=height calculation)

# stat height calculation:

# 1) 'identity' plots the value associated with each X category (use when
# graphing from a frequency table)

# 2) 'count' plots the number of cases at each x position (use when graphing
# observational data)

Scatter Plots

`+geom_point()`  # add points to a line chart

Additional Features

Regression Line

`+geom_smooth(method = "lm", se = FALSE, fullrange = TRUE)`  # Add Regression Line

# geom_smooth(method=’regression model’, se= to display the confidence
# interval, fullrange=to fit the plot or the data)

# SE =

# FALSE to not display the CI

# TRUE to display the CI

# fullrange =

# TRUE to fit the span to the range of the plot

# FALSE to fit the span to the range of the data
Vertical Line

+geom_vline(xintercept = 5, color = "blue")  # Add a vertical line

# geom_vline(xintercept=where the vertical line crosses the X-axis, 
# color='color of the line')

Error Bars

+geom_errorbar(ymin = (2), ymax = (4), width = 0.2)  # Add vertical error bars

# geom_errorbar(ymin=(lower error), ymax=(upper error), width=horizontal 
# width)

# ymin & ymax can also be calculations

Axis Options

Y-Axis

+scale_y_continuous(breaks = seq(1, 10, 1), expand = c(0, 0), limits = c(0, 
10.5))  # Change Y axis details

# scale_y_continuous(breaks=seq(minimum tick, maximum tick, major unit), 
# expand=c(multiplicative distance from X axis, additive distance from X 
# axis), limits = c(axis minimum, axis maximum))

X-Axis

+scale_x_continuous(breaks = seq(1, 10, 1), expand = c(0, 0), limits = c(0, 
10.5))  # Change X axis details

# scale_x_continuous(breaks=seq(minimum tick, maximum tick, major unit), 
# expand=c(multiplicative distance from X axis, additive distance from X 
# axis), limits = c(axis minimum, axis maximum))

Discrete X-Axis

+scale_x_discrete(limits = c("A", "B", "C"))

# changes the X-axis when the scale is not continuous

# scale_x_discrete(limits=c('labels in desired order for each X-axis 
# category'))
Y-Axis Label

```r
+ylab("Observed Y Values")  # Add Y-axis Label
```

```r
# ylab('Label for the Y-axis')
```

X-Axis Label

```r
+xlab("Observed X Values")  # Add X-axis Label
```

```r
# ylab('Label for the X-axis')
```

Additional Formatting

Plot Background

```r
+theme_bw()  # Change the plot background from grey to white
```

Text Formatting

```r
+theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotates X-axis text
```

```
# theme(axis.text.x = element_text(angle = label rotation in degrees, hjust
# = horizontal justification))

# horizontal justification:

# 0 means left-justified (where the text lines up)

# 1 means right-justified
```

Additional Text

```r
+annotate(geom = "text", x = 5, y = 5.5, label = , angle = 90, vjust = -0.5)
```

```
# Add text to a graph

# annotate(geom="text", x = x value at which to place the text, y = y value
# at which to place the text, label="text to be added", angle=rotation of
# the text, vjust=vertical justification)
```