### Basics

```
print("hello")
mynumber <- 5
# assigns value to object
myvector <- c(1,2,3)
# combines elements into vector
myvector[1]
# first element in vector
Fingers$Sex
# variable in data frame
```

### Tables

```
tally(myvector)
tally(~ Condition, data =
MindsetMatters)
tally(~ Thumb > 65, data = Fingers)
tally(Thumb ~ Sex, data = Fingers,
margins = TRUE, format =
"proportion")
```

### Fitting and Evaluating Models

```
Empty.model <- lm(Thumb ~ NULL,
data = Fingers)
Sex.model <- lm(Thumb ~ Sex, data
= Fingers)
Height.fun<- makeFun(Height.model)

predict(Empty.model)

resid(Empty.model)
anova(Empty.model)
supernova(Sex.model)
```

### Data

```
str(MindsetMatters)
head(MindsetMatters)
tail(MindsetMatters)
sort(myvector)
arrange(Fingers, Thumb)

select(Fingers, Sex, RaceEthnic,
Thumb)
       # selects variables
filter(Fingers, SSLast != "NA")
       # selects cases
head(select(Fingers, Thumb))

as.factor(Fingers$Sex)
as.numeric(Fingers$Interest)
factor(Fingers$Sex, levels =
c(1,2), labels = c("female",
"male"))
recode(Fingers$Job, "0" = 0, "1" =
50, "2" = 100)

ntile(Fingers$Height, 2)
# creates equal sized groups

aggregate(Happiness ~ Region, data
= HappyPlanetIndex, FUN = mean)
```

### Operators

```
sum(1,2,100)
+, -, *, /
>, <, >=, <=, ==, !=

Fingers$RingLonger <-
Fingers$Ring > Fingers$Index
# results in TRUE or FALSE

abs(Fingers$Residual)
Fingers$Residual^2
sqrt(157)
```

### Simple Statistics

```
mean(Fingers$Thumb)
var(Fingers$Thumb)
sd(Fingers$Thumb)
favstats(~ Wt, data = MindsetMatters)
cohensD(Thumb ~ Sex, data = Fingers)
cor(Thumb ~ Height, data = Fingers)
```

### Probability Distributions

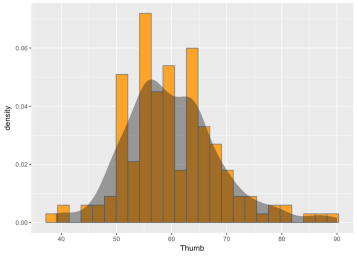
```
xpnorm(65.1, Thumb.stats$mean,
Thumb.stats$sd)

zscore(Fingers$Thumb)

qt(.975, df = 999)
# returns t at this probability
qf(.95, df1 = 1, df2 = 100)
# returns F at this probability

confint(Empty.model)
# CI using t distribution
```

### Simulation & Resampling

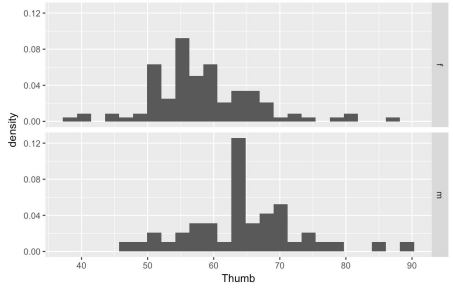
```
sample(Fingers$Thumb, 10)
# sample without replacement
resample(Fingers$Thumb, 157)
# sample with replacement
do(3) * resample(Fingers$Thumb,10)

shuffle(Servers$RandomGroups1)
# mixes up values in a variable

simThumb <- rnorm(10000,
Thumb.stats$mean, Thumb.stats$sd)
# simulates sampling from a normal
distribution

simPop <- data.frame(simThumb)

simSDoM <- do(10000) *
mean(rnorm(157, Thumb.stats$mean,
Thumb.stats$sd))
# simulates sampling distribution
of means
bootSDoM <- do(10000) *
mean(resample(Fingers$Thumb,157))
# bootstraps sampling distribution
of means
```

## Plots

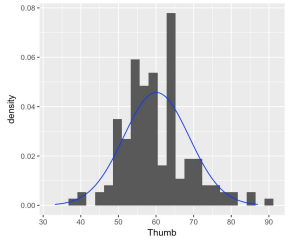
```
gf_histogram(~ Thumb, data = Fingers, color = "red", fill = "gray", bins = 10, binwidth = 4)

  %>% gf_labs(title= "Student Thumb Lengths", x= "Thumb Length (mm)")
# changes labels
  %>% gf_density()
# adds density curve to a histogram
```



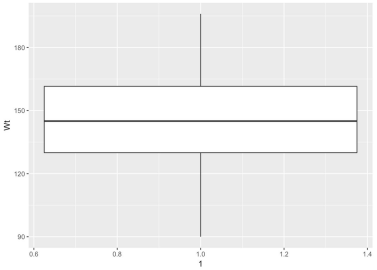

```
 gf_histogram(..density..~ Thumb, data = Fingers)
  %>% gf_facet_grid(Sex ~ .)
```



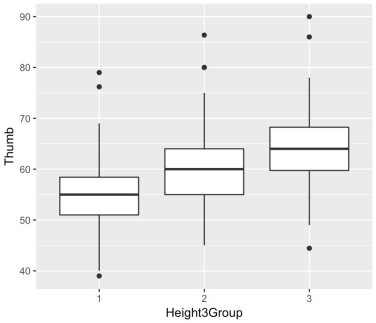

```
  %>% gf_dist("norm", color = "blue", params = list(Thumb.stats$mean, Thumb.stats$sd))
```



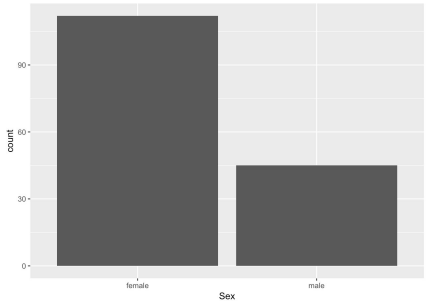

```
gf_boxplot(Wt ~ 1, data = MindsetMatters)
```



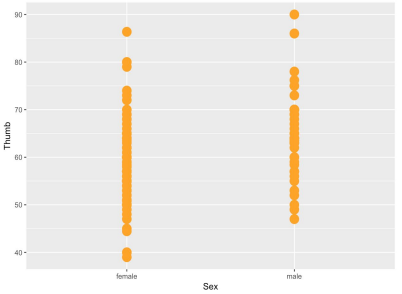

```
gf_boxplot(Thumb ~ Height3Group, data = Fingers)
```



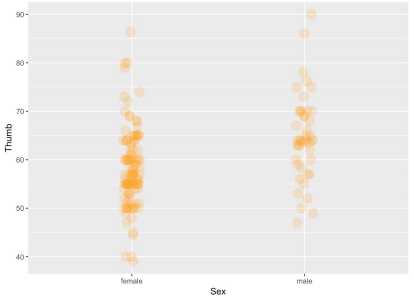

```
gf_bar( ~ Sex, data = Fingers)
# creates bar graph
```




```
gf_point(Thumb ~ Sex, data = Fingers, color = "orange", size = 5)
```




```
gf_jitter(Thumb ~ Sex, data = Fingers, color = "orange", size = 5, alpha = .5)
```




```
gf_point(Thumb ~ Height, data = Fingers, size = 4) %>%
  gf_lm(color = "orange")
# adds a regression line
```