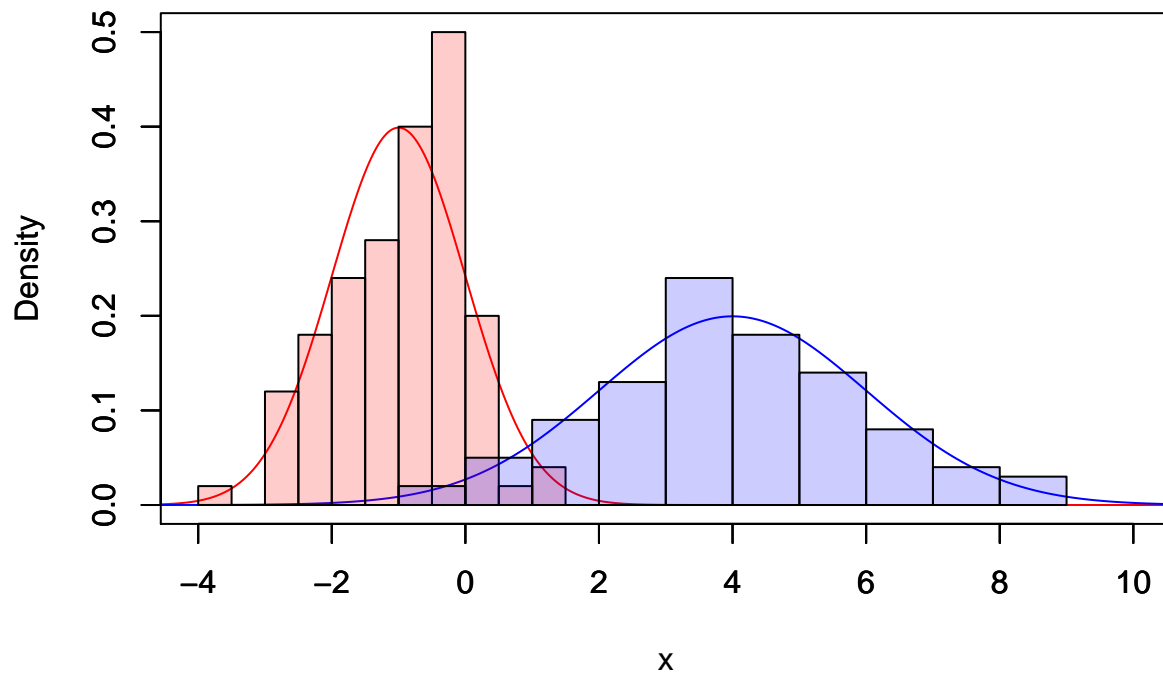


Hypothesis Testing, Part 1

Nicholas A. Del Grosso

July 23, 2018

Two Normal Population Distributions, with Sample Data (N=100)



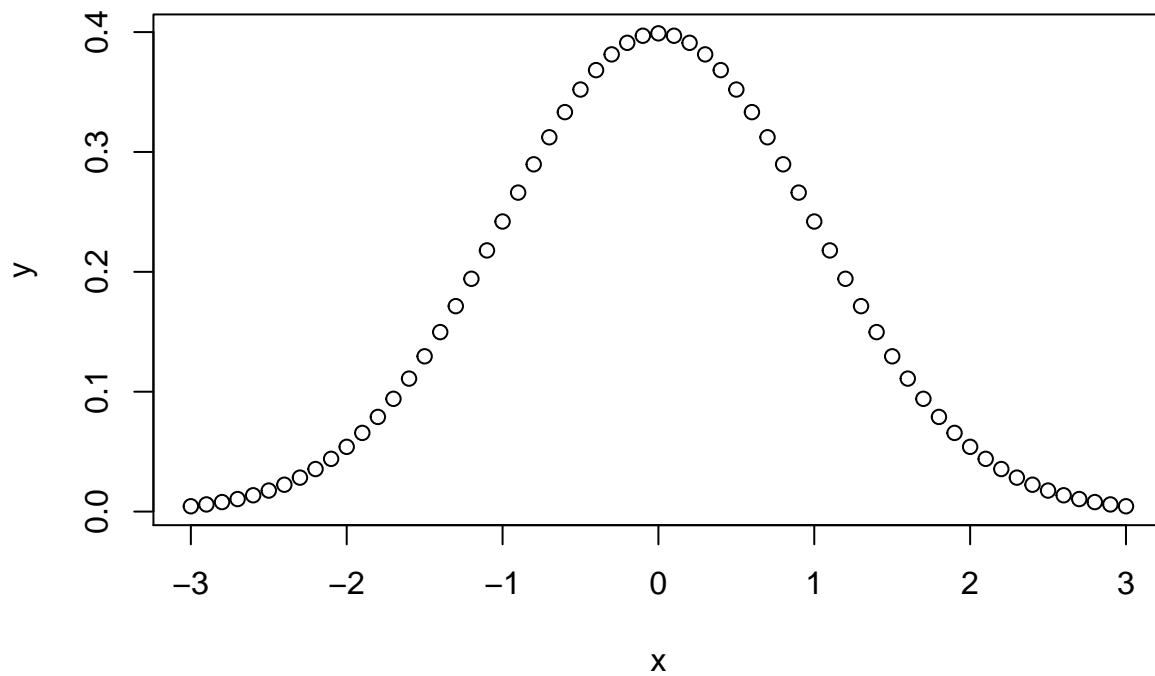
Querying and Generating Probability Densities

R's built-in stats package comes with lots of functions for working with common probability distributions. The functions we will review here allow you to:

- Get the distribution's **(d)ensity** height, given x values: `dnorm()`, `dpois()`, `dunif()`
- Get **(r)andom** amples from the distribution: `rnorm()`, `rpois()`, `runif()`
- Get the distributions's quantiles:
 - The **(q)uantile** of the distribution, given a percentile: `qnorm()`, `qpois()`, `qunif()`
 - The **(p)ercentile** of the distribution, given a quantile: `pnorm()`, `ppois()`, `punif()`

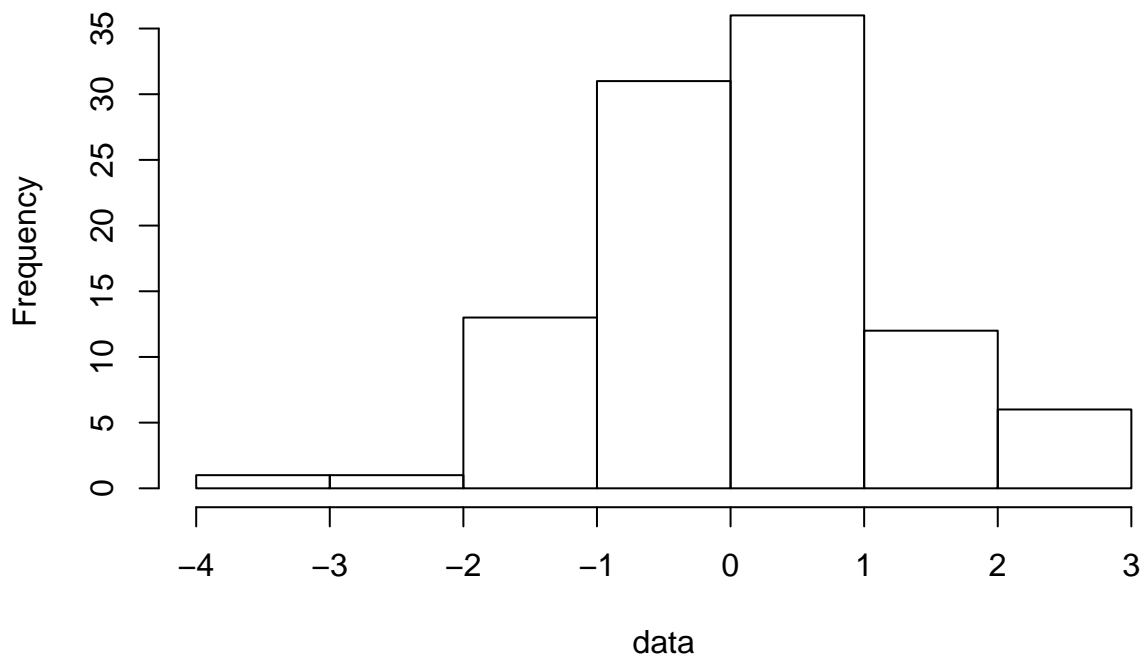
Examples of usage:

```
x <- seq(-3, 3, .1)
y <- dnorm(x, mean=0, sd=1)
plot(x, y)
```



```
data <- rnorm(100, mean=0, sd=1)
hist(data)
```

Histogram of data



```
qnorm(p=.05, mean=0, sd=1)
```

```
## [1] -1.644854
```

```
pnorm(q=-1.645, mean=0, sd=1)
```

```
## [1] 0.04998491
```

T-Tests

The `t.test()` function does t-tests. Setting different options in the function arguments lets you use it as one-sided or two-sided test, a paired-sample or independent-sample test, assume equal variance, etc. It returns a table of the results:

```
x <- rnorm(100, mean=-0.5, sd=1)
y <- rnorm(100, mean=0.5, sd=1)
t.test(x, y, alternative = "two.sided")

##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -5.9944, df = 197.76, p-value = 9.532e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   -1.0771268 -0.5438562
## sample estimates:
##  mean of x  mean of y
## -0.4332610  0.3772305
```

Nonparametric alternatives to the t-test are also available: - Kruskal-Wallis Test: `kruskal.test()` - Mann-Whitney U Test: `wilcox.test()` - Wilcoxon Rank Sum Test: `wilcox.test(paired=TRUE)`

If you save the result of the test as a variable, you can also access the details of the table and index the **names** of the result to access the data:

```
results = t.test(x, y, alternative = "two.sided")
names(results)

## [1] "statistic"    "parameter"    "p.value"      "conf.int"     "estimate"
## [6] "null.value"   "alternative"   "method"       "data.name"

results['p.value'] # Can also be written as: results$p.value

## $p.value
## [1] 9.532029e-09
```

Correlations

Testing for correlations between paired samples can be done with the `cor.test()` function. By default it does a Pearson correlation, but can also do a Spearman or Kendall's Tau test by changing the “method” argument

Exercises

Exercises are available at the `hypothesis_testing_exercises.R` script.