R supports a large number of distributions. Usually, four types of functions are provided for each distribution:

- **d*: density function
- **p*: cumulative distribution function, \( P(X \leq x) \)
- **q*: quantile function
- **r*: draw random numbers from the distribution

* represents the name of a distribution.
Probability distributions

Density

CDF

X

Probability distributions

Density

CDF

X
The distributions supported include continuous distributions:

- **unif**: Uniform
- **norm**: Normal
- **t**: t
- **chisq**: Chi-square
- **f**: F
- **gamma**: Gamma
- **exp**: Exponential
- **beta**: Beta
- **lnorm**: Log-normal
As well as discrete ones:

- **binom**: Binomial
- **geom**: Geometric
- **hyper**: Hypergeometric
- **nbinom**: Negative binomial
- **pois**: Poisson
Examples of using these functions: Generate 5 random numbers from $N(2, 2^2)$.
Generate 5 random numbers from $N(2, 2^2)$

```r
> rnorm(5, mean=2, sd=2)
[1] 5.4293122 -0.6731407 -1.1743455 1.5155376 -0.3100879
```
Obtain 95% quantile for the standard normal distribution
Obtain 95% quantile for the standard normal distribution

> qnorm(0.95)
[1] 1.644854
Compute cumulative probability \( Pr(X \leq 3) \) for \( X \sim t_5 \) (i.e. t-distribution, d.f. = 5)
Compute cumulative probability $Pr(X \leq 3)$ for $X \sim t_5$ (i.e. t-distribution, d.f.=5)

> pt(3, df=5)
[1] 0.9849504
Compute one-sided p-value for t-statistic $T=3$, d.f.=$5$
Compute one-sided p-value for t-statistic $T=3$, d.f.$=5$

```r
> pt(3, df=5, lower.tail=FALSE)
[1] 0.01504962
```
Plot density function for beta distribution Beta(7,3)
Probability distributions

Plot density function for beta distribution Beta(7,3)

```r
> x<-seq(0,1,by=0.01)
> y<-dbeta(x,7,3)
> plot(x,y,type="l")
```
There are three types of t-test:

- one-sample t-test
- two-sample t-test
- paired t-test
One sample t-test

Histogram of x

```
                x
Frequency
−4 −2  0  2  4
0  2  4  6  8
```

R: Statistical Functions
One sample t-test

Data: $x_1, \ldots, x_n$

Assumptions: $x_i \overset{i.i.d.}{\sim} N(\mu, \sigma^2)$.

Question: Is $\mu$ equal to $\mu_0$?
One sample t-test

Now perform test:

1. Hypotheses: \( H_0 : \mu = \mu_0 \) vs. \( H_1 : \mu \neq \mu_0 \)

2. Test statistic: \( T_{obs} = \frac{\bar{X} - \mu_0}{SE(\bar{X})} \) where \( SE(\bar{X}) = \frac{s}{\sqrt{n}} \) and
   \[
s = \sqrt{\frac{\sum_i (x_i - \bar{x})^2}{n-1}}
   \]

3. Degrees of freedom: \( d.f. = n - 1 \)

4. p-value: one-sided = \( Pr( T_{d.f.} \geq T_{obs} ) \) (or \( Pr( T_{d.f.} \leq T_{obs} ) \));
   two-sided = \( Pr( | T_{d.f.} | \geq | T_{obs} | ) \)

5. Confidence interval: \( (1 - \alpha) \) CI = \( \bar{X} \pm t_{d.f.} (1 - \alpha/2) \times SE(\bar{X}) \)
t.test(x, y = NULL,
    alternative = c("two.sided", "less", "greater"),
    mu = 0, paired = FALSE, var.equal = FALSE,
    conf.level = 0.95, ...)
> t.test(z)

     One Sample t-test

data:  z
t = 1.9453, df = 5, p-value = 0.1093
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
  -0.1808551  1.3060859
sample estimates:
  mean of x
0.5626154
```r
> u <- t.test(z)
> summary(u)

<table>
<thead>
<tr>
<th></th>
<th>Length</th>
<th>Class</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>statistic</td>
<td>1</td>
<td>-none-</td>
<td>numeric</td>
</tr>
<tr>
<td>parameter</td>
<td>1</td>
<td>-none-</td>
<td>numeric</td>
</tr>
<tr>
<td>p.value</td>
<td>1</td>
<td>-none-</td>
<td>numeric</td>
</tr>
<tr>
<td>conf.int</td>
<td>2</td>
<td>-none-</td>
<td>numeric</td>
</tr>
<tr>
<td>estimate</td>
<td>1</td>
<td>-none-</td>
<td>numeric</td>
</tr>
<tr>
<td>null.value</td>
<td>1</td>
<td>-none-</td>
<td>numeric</td>
</tr>
<tr>
<td>alternative</td>
<td>1</td>
<td>-none-</td>
<td>character</td>
</tr>
<tr>
<td>method</td>
<td>1</td>
<td>-none-</td>
<td>character</td>
</tr>
<tr>
<td>data.name</td>
<td>1</td>
<td>-none-</td>
<td>character</td>
</tr>
</tbody>
</table>
```
Two sample t-test

Histogram of $x$

Histogram of $y$
Two sample t-test

Data: $x_1, \ldots, x_m; y_1, \ldots, y_n$

Assumptions: $x_i \overset{i.i.d.}{\sim} N(\mu_1, \sigma_1^2); y_i \overset{i.i.d.}{\sim} N(\mu_2, \sigma_2^2)$

Question: Is $\mu_1 - \mu_2$ equal to $d$?
Two sample t-test

Perform test if $\sigma^2_1 = \sigma^2_2$:

1. **Hypotheses:** $H_0: \mu_1 - \mu_2 = d$ vs. $H_1: \mu_1 - \mu_2 \neq d$

2. **Test statistic:** $T_{obs} = \frac{\bar{X} - \bar{Y} - d}{SE(\bar{X} - \bar{Y})}$ where $SE(\bar{X} - \bar{Y}) = s_p \sqrt{\frac{1}{m} + \frac{1}{n}}$ and
   \[ s_p = \sqrt{\frac{(m-1)s^2_X + (n-1)s^2_Y}{m+n-2}} \]

3. **Degrees of freedom:** $d.f. = m + n - 2$

4. **p-value:** one-sided = $Pr(T_{d.f.} \geq T_{obs})$ (or $Pr(T_{d.f.} \leq T_{obs})$); two-sided = $Pr(|T_{d.f.}| \geq |T_{obs}|)$

5. **Confidence interval:**
   \[(1 - \alpha) \ CI = (\bar{X} - \bar{Y}) \pm t_{d.f.}(1 - \alpha/2) \times SE(\bar{X} - \bar{Y})\]
Two sample t-test

Perform test if $\sigma_1^2 \neq \sigma_2^2$:

1. Test statistic: $T_{obs} = \frac{\bar{X} - \bar{Y} - d}{SE(\bar{X} - \bar{Y})}$ where $SE(\bar{X} - \bar{Y}) = \sqrt{\frac{s_X^2}{m} + \frac{s_Y^2}{n}}$

2. Degrees of freedom (Welch-Satterthwaite approximation):

$d.f. = \frac{(\frac{s_X^2}{m} + \frac{s_Y^2}{n})^2}{\frac{s_X^4}{m^2(m-1)} + \frac{s_Y^4}{n^2(n-1)}}$
Example:

```r
> x<-rnorm(10,1,1)
> y<-rnorm(15,2,1)
> t.test(x,y)
```

```
Welch Two Sample t-test

data:  x and y
t = -4.1207, df = 22.099, p-value = 0.0004458
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  -1.7046928 -0.5634708
sample estimates:
mean of x  mean of y
  1.136442   2.270524
```
Paired t-test

Data: $x_1, \ldots, x_n; y_1, \ldots, y_n$; $x_i$ and $y_i$ are paired

Assumptions: $(x_i - y_i)^i \sim N(\mu, \sigma^2)$

Essentially the same as one-sample t-test.
Simple Linear Regression

Data: \((y_1, x_1), \ldots, (y_n, x_n)\)

Assumption: \(Y|X \sim N(\beta_0 + \beta_1 X, \sigma^2)\)

There are several different questions one can ask:

- What are \(\beta_0\) and \(\beta_1\)? Are they different from zero?
- How much information does \(X\) have for explaining variations in \(Y\)?
- Given a new \(x\), what is the predicted value of \(y\)?

In order to answer them, you will need to find out what \(\beta_0\) and \(\beta_1\) are.
Least squares estimates are estimates of $\beta_0$ and $\beta_1$ that minimize $\sum_i (y_i - \beta_0 - \beta_1 x_i)^2$.

The solution to this minimization is:

- $\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$
- $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$

$\epsilon_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i$ is called residual.

$\hat{\sigma} = \sqrt{\frac{\sum_i \epsilon_i^2}{d.f.}}$

$d.f. = n - (\text{no. of regression coefficients}) = n - 2$
Simple Linear Regression

\[ SE(\hat{\beta}_1) = \hat{\sigma} \sqrt{\frac{1}{(n-1)s_X^2}}, \text{ d.f. } = n - 2 \]

\[ SE(\hat{\beta}_0) = \hat{\sigma} \sqrt{\frac{1}{n} + \frac{\bar{X}^2}{(n-1)s_X^2}}, \text{ d.f. } = n - 2 \]

T-test can be used to test whether coefficients are significantly different from zero.
In R, you can use `lm()` to fit this *linear model*.

For example:

```r
> x <- rnorm(16, mean=3, sd=2)
> y <- 0.2 + 0.1 * x + rnorm(16, mean=0, sd=0.3)
> z <- lm(y ~ x)
> summary(z)
```

Call:
`lm(formula = y ~ x)`

Residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-0.65999</td>
<td>-0.27410</td>
<td>0.01021</td>
<td>0.27423</td>
<td>0.53585</td>
</tr>
</tbody>
</table>

Coefficients:

|                  | Estimate | Std. Error | t value | Pr(>|t|) |
|------------------|----------|------------|---------|---------|
| (Intercept)      | 0.28748  | 0.14855    | 1.935   | 0.0734  |
| x                 | 0.05696  | 0.05153    | 1.105   | 0.2877  |

Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1  1

Residual standard error: 0.3594 on 14 degrees of freedom
Multiple R-squared: 0.08025,   Adjusted R-squared: 0.01456
F-statistic: 1.222 on 1 and 14 DF,  p-value: 0.2877
`lm()` returns an object of class “`lm`”. It is a list containing the following components:

- **coefficients**: a named vector of coefficients
- **residuals**: the residuals, that is response minus fitted values.
- **fitted.values**: the fitted mean values.
- **rank**: the numeric rank of the fitted linear model.
- **weights**: (only for weighted fits) the specified weights.
- **df.residual**: the residual degrees of freedom.
- ...
Simple Linear Regression

-1 0 1 2 3 4 5
0.0 0.2 0.4 0.6 0.8 1.0
x

Normal Q-Q Plot
Theoretical Quantiles
Sample Quantiles

z$fitted
z$res

-2 -1 0 1 2
-0.6 -0.4 -0.2 0.0 0.2 0.4

140.776 Statistical Computing
R: Statistical Functions
$R^2 = 1 - \frac{\sum_i \epsilon_i^2}{\sum_i (y_i - \bar{y})^2}$

$= 100 \times \left( \frac{\text{Total sum of squares} - \text{Residual sum of squares}}{\text{Total sum of squares}} \right) \%$

R-squared tells you what fraction of variance in the response variable $Y$ is explained by covariate $X$. 
It is easier to interpret the simple linear regression if you rewrite it in the following form:

\[ Y - \bar{Y} = r \frac{\hat{\sigma}_Y}{\hat{\sigma}_X} (X - \bar{X}) \]

Also,

\[ R - squared = r^2 \] where \( r \) is sample correlation coefficient.
Multiple Regression

Simple linear regression can be generalized to have multiple covariates:

\[ Y | X_1, \ldots, X_m \overset{\text{ind.}}{\sim} N(\beta_0 + \beta_1 X_1 + \ldots + \beta_m X_m, \sigma^2) = N(X\beta, \sigma^2) \]

Least square estimates for \( \beta \) are:

\[ \hat{\beta} = (X^T X)^{-1} X^T Y \]
Multiple Regression

For example:

```r
> fit2<-lm(z~x+y)
> summary(fit2)

Call:
  lm(formula = z ~ x + y)

Residuals:
     Min       1Q   Median       3Q      Max
-2.75339 -0.62698  0.08483  0.61041  2.08833

Coefficients:
               Estimate Std. Error t value  Pr(>|t|)
(Intercept)       0.09939   0.20922   0.475   0.636
x                 0.96199   0.09292  10.353  <2e-16 ***
y                1.93263   0.09402  20.556  <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 0.9889 on 97 degrees of freedom
Multiple R-squared: 0.842, Adjusted R-squared: 0.8387
F-statistic: 258.4 on 2 and 97 DF, p-value: < 2.2e-16
```
glm() can be used to handle generalized linear models.

\[
\text{glm(formula, family = gaussian, data, weights, subset,}
\]
\[
\text{na.action, start = NULL, etastart, mustart,}
\]
\[
\text{offset, control = glm.control(...), model = TRUE,}
\]
\[
\text{method = "glm.fit", x = FALSE, y = TRUE, contrasts = NULL,}
\]
\[
\text{...)}
\]