**Basics**

* Basic types: logical, integer, real, complex, character, raw.
* Basic mathematical and logical operations (+, -, /, \*, %%, |, &, !).
* Create/print/remove new variables. Get/change variable types.
* c, length.
* NULL, NA, NaN.
* Flow control (if statements, for loops).
* Numerical and logical indexing.
* Use functions (vectorization, default values) and write custom functions.
* Use built-in help, install packages, source other R files, set/get working directory.

**Vectors and matrices**

* Create vectors and matrices.
* Get dimensionality of vectors and matrices.
* Basic matrix algebra: matrix addition, multiplication, transpose, determinant, trace, eigenvalue decomposition, Cholesky decomposition, sub-matrices, matrix diagonal.

**Other types & data structures**

* **factor:** Create ordered and unordered factors. Get the levels of a factor, rename levels of a factor, drop unused levels of a factor.
* **data.frame:** Create a data frame.Rename columns.Modify, add and remove columns and rows.
* **list:** Create a list.Name and rename elements.Modify, add and remove list elements.

**I/O**

* Read and write data in CSV format (with/without column/row names, ).
* Read and write serialized R objects (saveRDS, readRDS).

**Probability distributions**

* Use the PDF/PMF, CDF, quantile function and generate random variates for all distributions covered in this course that are supported in base R.
* set.seed.

**Statistical graphics (use of ggplot2 recommended but not mandatory)**

* Basic plots: scatterplot, lineplot, boxplot, histogram, density plot.
* Adding to a plot: title, key/legend, text annotation, lines.

**Useful built-in functions**

* Working with strings (grep, gsub, paste, paste0, substr).
* Creating sequences of values (rep, seq).
* Rounding (round, floor, ceil).
* Numerical summaries (mean, median, var, sd, quantile, min, max).
* Apply functions (apply, tapply, lapply, sapply).
* Other (which, unique, table, sort, order, rank, sample).

Example task 1:

1. Create a sequence of numbers from -5 to +5 with 0.01 increment.
2. Plot the density curve of the normal N(1, 3) distribution over the range from (a).
3. Store the sequence and density value in a data.frame.
4. Save the serialized data.frame to a file.

Example task 2:

1. Create a sequence of length 100 such that every element of the sequence is uniformly drawn at random between 1 and 12.
2. Create a factor from the sequence from (a).
3. Rename the levels of the factor from (b) from 1, 2, ..., 12 to Jan, Feb, ..., Dec.
4. Sort the sequence.

Example task 3:

1. Create a sequence of length 100 such that every element of the sequence is uniformly drawn at random between 1 and 12.
2. Print the number that appears the most times in the sequence.
3. Write a *mode(x)* function. That is, a function that returns the most common element in the sequence. If two or more elements are tied, it should return all of them.

Example task 4:

1. Create a data.frame with 100 rows and 2 columns X and Y. Draw the values of X and Y from a beta and gamma distribution, respectively.
2. Save the data.frame to csv file (no row numbers, use semicolon as separator).
3. Load the data.frame from file.
4. Plot a scatterplot of X and Y.

Example task 5:

1. Load a serialized list object from xyz.rds.
2. Print the number of elements in the list.
3. Remove the 2nd element from the list.
4. Remove the element named »blue« from the list.
5. Add an element called »red« to the list with value 5.

Example task 6:

1. Create 100 random strings of length 20 using characters from a to g.
2. Find the string with the most occurences of the substring »bag«.
3. Change all occurences of »bag« in the sequence from (b) with »bed«.

Example task 7:

1. Create a m x n matrix A (m < n) with elements drawn uniformly at random from the interval [-1, 1].
2. Create a square matrix B = ATA.
3. Find the eigenvalues of B.