Foundations of Data Science for Biologists

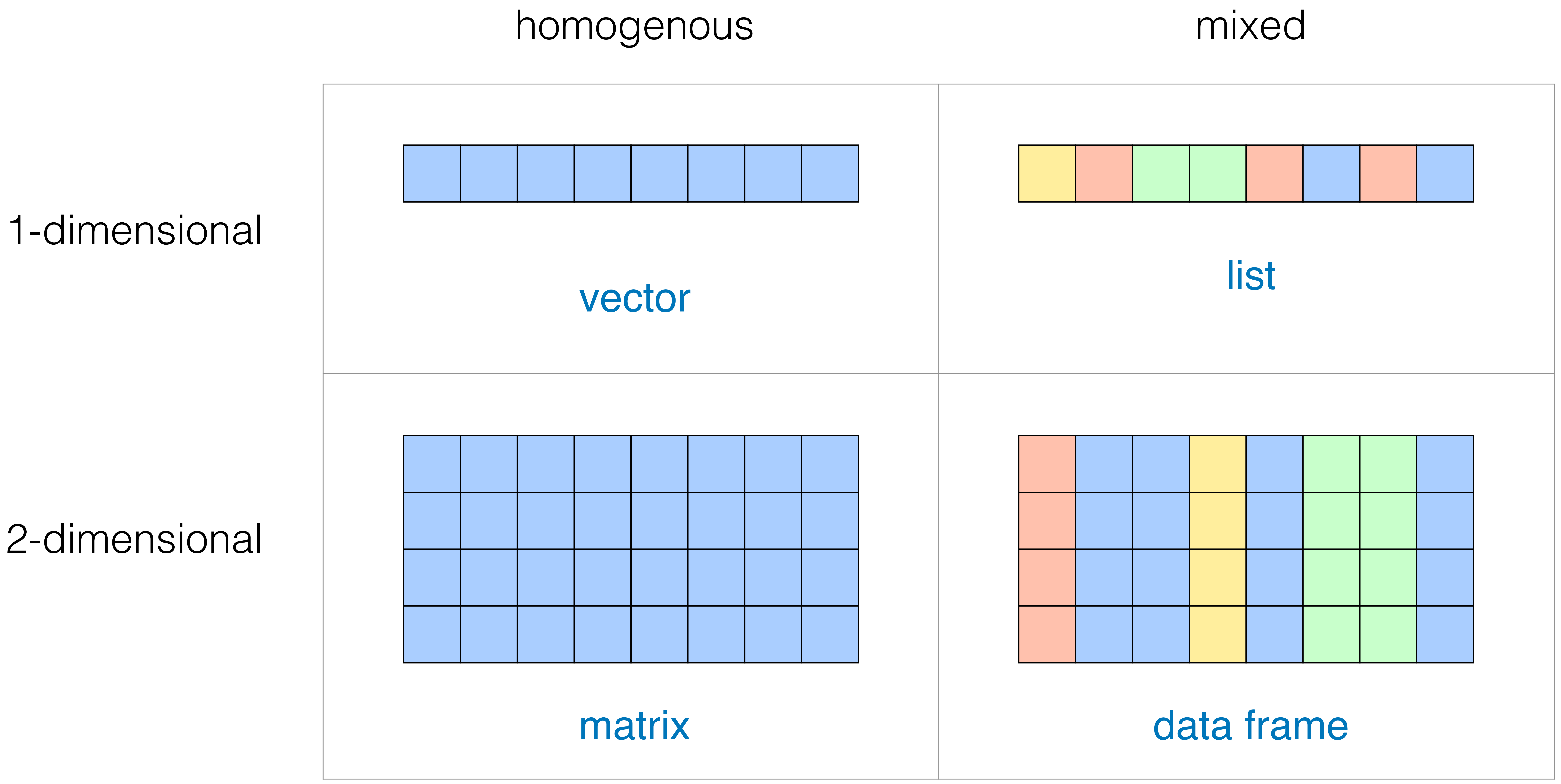
R: data types and indexing

BIO 724D

02-SEP-205

Instructors: Paul Magwene, Greg Wray, Kayla Wilhoit

Taxonomy of basic data types in R



Data types

Computers work with 0s and 1s — but you want to work with numbers, names, dates, etc.

Data types instruct programs how to interpret and process different kinds of data

- Common data types in R are integer, float, character, and logical

R has an extensive set of rules for each data type:

- What **values** are allowed (e.g., an integer can be `42` but not `'42'` or `42.7`)
- What **operations** are allowed (e.g., division for float but not character or logical)
- How to **display** data in human-readable form (e.g., `01010010` as **R** or **82**)

Data objects

When you assign a value to a variable name, R stores information in two places:

A **data object** that contains:

- **Data:** values, such as `-23.84` or `'Adelie'`
- **Metadata:** the data type, how many values, etc.

A table containing the names and address of all variables currently in use

When you type a variable name R:

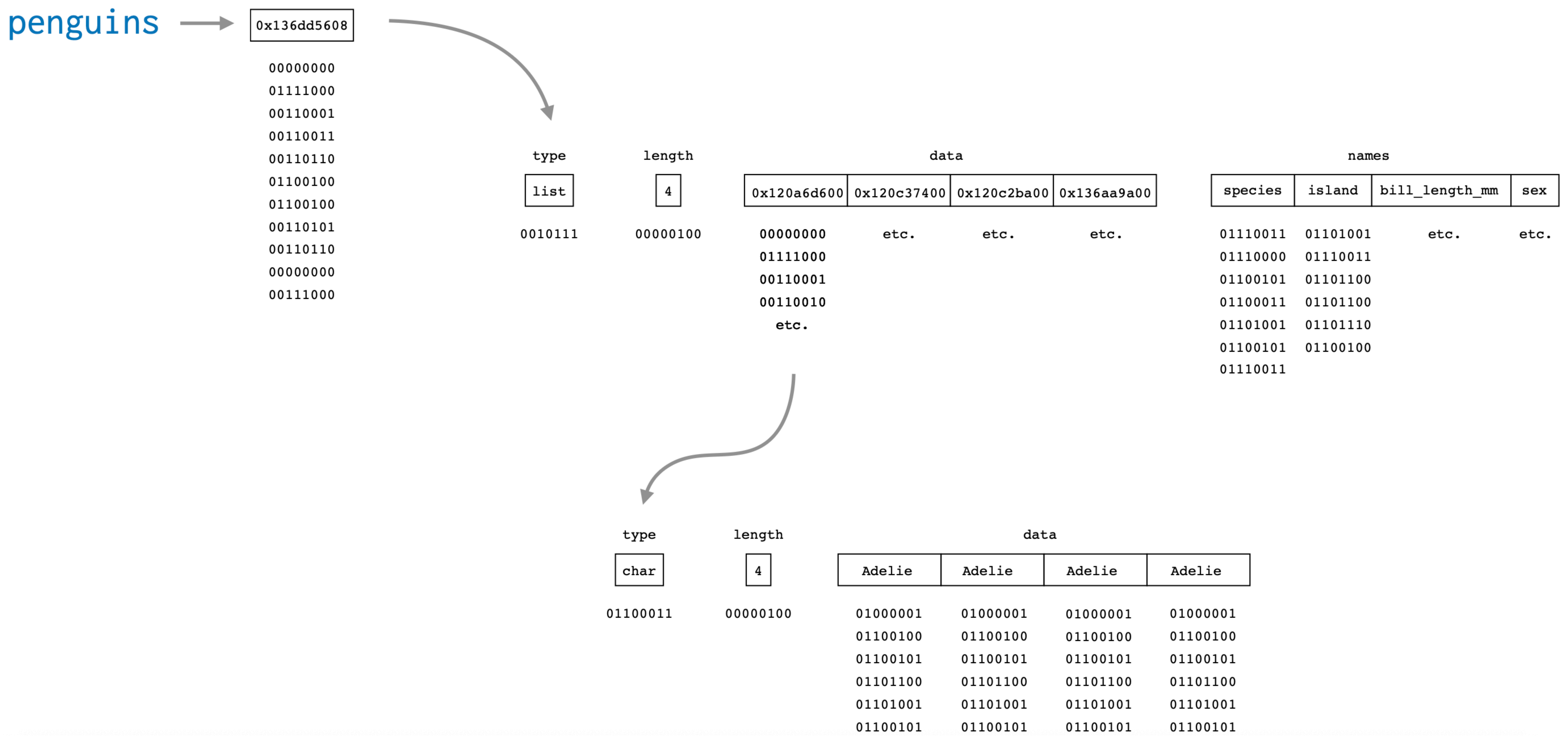
- Retrieves the address of the data object
- Knows how to interpret the data (as numbers, letters, etc.), how many values, etc.

What a data frame looks like to you

penguins →

species	island	bill_length_mm	sex
Adelie	Torgersen	39.1	male
Adelie	Torgersen	39.5	female
Adelie	Biscoe	37.8	female
Adelie	Biscoe	37.7	male

What a data frame looks like to R



Converting between data types

It is often possible and useful to convert between data types (called **coercion** in R)

- Must be a homogenous data type (vector, matrix, or column in a data frame)
- Must make logical sense (e.g., "2" can be coerced to integer but "kangaroo" cannot)

To coerce, use `as.integer()`, `as.logical()`, `as.character()`, etc.

Coercion rules to be aware of:

Numeric to integer	truncates any decimal values (does not round!)
Numeric to logical	`0` becomes `FALSE`; non-zero values become `TRUE`
Logical to numeric	`TRUE` becomes `1`, `FALSE` becomes `0`
Numeric to character	numerals and symbols become characters
Character to numeric	must be a formatted number (`-`, `+` and `.` allowed)

And many more; check documentation to avoid unexpected results!

Missing values

R provides three special values that represent missing, invalid, or undefined information

`NA`	a missing value; acronym for not available
`NaN`	an invalid mathematical result (e.g., `0/0`); acronym for not a number
`NULL`	a value that is undefined (e.g. vector of length `0`)

Points to remember:

Do not use quotes: `‘NA’` is interpreted a character value

Do not use in mathematical operations: `my_var + NA` substitutes every item with `NA`

Do not use in logical tests: `my_var == NA` returns `NA`

To identify missing values:

`is.na(my_vec)`	returns a logical vector with NAs FALSE, all others TRUE
`which(is.na(my_vec))`	returns the position(s) of any NAs in the vector