Colors vs Real-World Objects

Basic Setup

```
## -- Attaching packages ---------------------------------------------------------------------------
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## Loading required package: lattice
## Loading required package: ggformula
## Loading required package: ggstance
##
## Attaching package: 'ggstance'
## The following objects are masked from 'package:ggplot2':
##
##     geom_errorbarh, GeomErrorbarh
##
## New to ggformula?  Try the tutorials:
##  learnr::run_tutorial("introduction", package = "ggformula")
##  learnr::run_tutorial("refining", package = "ggformula")
## Loading required package: mosaicData
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by this.
##
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.
##
## Have you tried the ggformula package for your plots?
```

```
##
## Attaching package: 'mosaic'

## The following object is masked from 'package:Matrix':
##
##     mean

## The following objects are masked from 'package:dplyr':
##
##     count, do, tally

## The following object is masked from 'package:purrr':
##
##     cross

## The following object is masked from 'package:ggplot2':
##
##     stat

## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##     quantile, sd, t.test, var

## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum

##
## Attaching package: 'scales'

## The following object is masked from 'package:mosaic':
##
##     rescale

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor

## Loading required package: knitr
```

Import Colors

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
```

```
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
```

```
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
```

```
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
```

```
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
```

```
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
```

```
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
```

```
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
```

```
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
```

```
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   change = col_double(),
##   responses = col_double(),
##   time = col_double(),
##   prev_condition = col_double(),
##   prev_change = col_double(),
##   color = col_double()
## )
```

```
## `summarise()` regrouping output by 'subject', 'age' (override with `.groups` argument)
```

Import Objects

```
## Warning in rm(objectList): object 'objectList' not found
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
```

```
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
```

```
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
```

```
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
```

```
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
```

```
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
```

```
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
```

```
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
```

```
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
```

```
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )

## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]

## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
```

```
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
```

```
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
```

```
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Duplicated column names deduplicated: 'time' => 'time_1' [9]
```

```
## Parsed with column specification:
## cols(
##   `#subject` = col_double(),
##   block = col_double(),
##   trial = col_double(),
##   condition = col_double(),
##   setsize = col_double(),
##   change = col_double(),
##   time = col_double(),
##   responses = col_double(),
##   time_1 = col_double(),
##   Pic = col_character(),
##   prev_condition = col_double(),
##   prev_setsize = col_double(),
##   prev_change = col_double(),
##   prev_time = col_double()
## )
```

```
## Warning: Problem with `mutate()` input `pic_num`.
## i NAs introduced by coercion
## i Input `pic_num` is `as.numeric(Pic)`.
```

```
## Warning in mask$eval_all_mutate(dots[[i]]): NAs introduced by coercion
```

```
## `summarise()` regrouping output by 'subject', 'age' (override with `.groups` argument)
```

```
## `summarise()` regrouping output by 'subject', 'age', 'setsize' (override with `.groups` argument)
```

```
## `summarise()` regrouping output by 'subject' (override with `.groups` argument)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
## `summarise()` ungrouping output (override with `.groups` argument)
## `summarise()` ungrouping output (override with `.groups` argument)
```

join color & objects

```
colorListResult %>%ungroup() %>%  select(age,condition,mean_answer_correct,mean_response_time,kPashler,k
names(ColorJoin) <- c("age_c","condition_c","mean_answer_correct_c","mean_response_time_c","kPashler_c"
objectResult %>% inner_join(ColorJoin,by = "joinKey")  %>%
  mutate(diffAnswerRate = mean_answer_correct-mean_answer_correct_c)%>%
  mutate(diffK = k-k_c)-> testsJoined
objectResultCondition %>% inner_join(ColorJoin,by = "joinKey") -> testConditionJoined
colorListResult %>% ungroup() %>% select(subject,age,condition,mean_answer_correct,mean_response_time,kl
```

```
objectResult %>%  ungroup() %>% select(subject,age,setsize,mean_answer_correct,mean_response_time,kPashl
joinedTestList = rbind(ObjectJoinLong,ColorJoinLong)
```

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

Erster Schritt Überblick der Gesamtanzahl der falschen und richtigen Antworten

```
## Warning: `count_()` is deprecated as of dplyr 0.7.0.
## Please use `count()` instead.
## See vignette('programming') for more help
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.

Gib eine Tabelle mit den zusammenfassenden Daten für Color & Objects aus

```
colorList %>% group_by(condition,age) %>% summarise(Gesamt = n(),Richtig = sum(answer_correct == 1),Fals
```

```
## `summarise()` regrouping output by 'condition' (override with `.groups` argument)
```

Table 1: Überblick Versuchsreihe Color

SetSize	age	Gesamt	Richtig	Falsch	ProzentRichtig	ProzentFalsch
2	old	1297	1226	71	95%	5%
2	young	1298	1268	30	98%	2%
4	old	1297	945	352	73%	27%
4	young	1303	1139	164	87%	13%
6	old	1291	863	428	67%	33%
6	young	1295	946	349	73%	27%
8	old	1315	782	533	59%	41%
8	young	1304	838	466	64%	36%

```
objectList %>% group_by(setsize,age) %>% summarise(Gesamt = n(),Richtig = sum(answer_correct == 1),Falsc
```

```
## `summarise()` regrouping output by 'setsize' (override with `.groups` argument)
```

Table 2: Überblick Versuchsreihe Real world objects

setsize	age	Gesamt	Richtig	Falsch	ProzentRichtig	ProzentFalsch
2	old	3840	3539	301	92%	8%
2	young	3840	3607	233	94%	6%
4	old	3840	3066	774	80%	20%
4	young	3840	3249	591	85%	15%
6	old	3840	2678	1162	70%	30%

setsize	age	Gesamt	Richtig	Falsch	ProzentRichtig	ProzentFalsch
6	young	3840	2911	929	76%	24%

```
#colorListResult %>% ungroup() %>% group_by(age,condition) %>% summarise(mean(hitRate),mean(falseAlarmR

colorResultGroupAge %>% select(age,SetSizeTotal,SetSize2,SetSize4,SetSize6,K_Total_mean,K_2_mean,K_6_mea
df_Color_Compress[2,-1] - df_Color_Compress[1,-1] -> df_Color_diff

colorResultGroupAge %>% select(age,SetSizeTotal,SetSize2,SetSize4,SetSize6,K_Total_mean,K_2_mean,K_6_mea
```

Table 3: Überblick Color Objekte Mittelwerte korrekte Antworten und K-Werte

age	SetSizeTotal	SetSize2	SetSize4	SetSize6	K_Total_mean	K_2_mean	K_6_mean	K_PTotal_mean	K_P2_mean	K_P4_mean
old	0.73	0.95	0.73	0.67	1.78	1.78	2.02	3.15	1.88	2.85
young	0.81	0.98	0.87	0.73	2.48	1.91	2.76	4.00	1.97	3.77

```
objectResultGroupAge %>% select(age,SetSizeTotal,SetSize2,SetSize4,SetSize6,K_Total_mean,K_2_mean,K_6_me
```

Table 4: Überblick Real World Objekte Mittelwerte korrekte Antworten und K-Werte

age	SetSizeTotal	SetSize2	SetSize4	SetSize6	K_Total_mean	K_2_mean	K_6_mean	K_PTotal_mean	K_P2_mean	K_P4_mean
old	0.81	0.92	0.80	0.70	2.15	1.69	2.37	2.80	1.74	2.89
young	0.85	0.94	0.85	0.76	2.54	1.76	3.10	3.17	1.82	3.22

```
objectResultGroupAge %>% select(age,SetSizeTotal,SetSize2,SetSize4,SetSize6,K_Total_mean,K_2_mean,K_6_me

df_Object_Compress[2,-1] - df_Object_Compress[1,-1] -> df_Object_diff

df_Color_diff %>% round(2) %>% kable(caption = "Abstand der Ergebnisse bei Color Objekten zwischen älte
```

Table 5: Abstand der Ergebnisse bei Color Objekten zwischen älteren Erwachsenen und jüngeren Erwachsenen

SetSizeTotal	SetSize2	SetSize4	SetSize6	K_Total_mean	K_2_mean	K_6_mean	K_PTotal_mean	K_P2_mean	K_P4_mean
0.07	0.03	0.15	0.06	0.7	0.13	0.74	0.85	0.09	0.92

```
df_Object_diff %>% round(2) %>% kable(caption = "Abstand der Ergebnisse bei Real World Objekten zwische
```

Table 6: Abstand der Ergebnisse bei Real World Objekten zwischen älteren Erwachsenen und jüngeren Erwachsenen

SetSizeTotal	SetSize2	SetSize4	SetSize6	K_Total_mean	K_2_mean	K_6_mean	K_PTotal_mean	K_P2_mean	K_P4_mean
0.04	0.02	0.05	0.06	0.39	0.07	0.73	0.37	0.08	0.33

```
df_Object_diff - df_Color_diff -> df_AltvsJung

df_AltvsJung %>% round(2) %>% kable(caption = "Abstand der Ergebnisse bei Color vs. Real World Objekten
```

Table 7: Abstand der Ergebnisse bei Color vs. Real World Objekten zwischen älteren Erwachsenen und jüngeren Erwachsenen

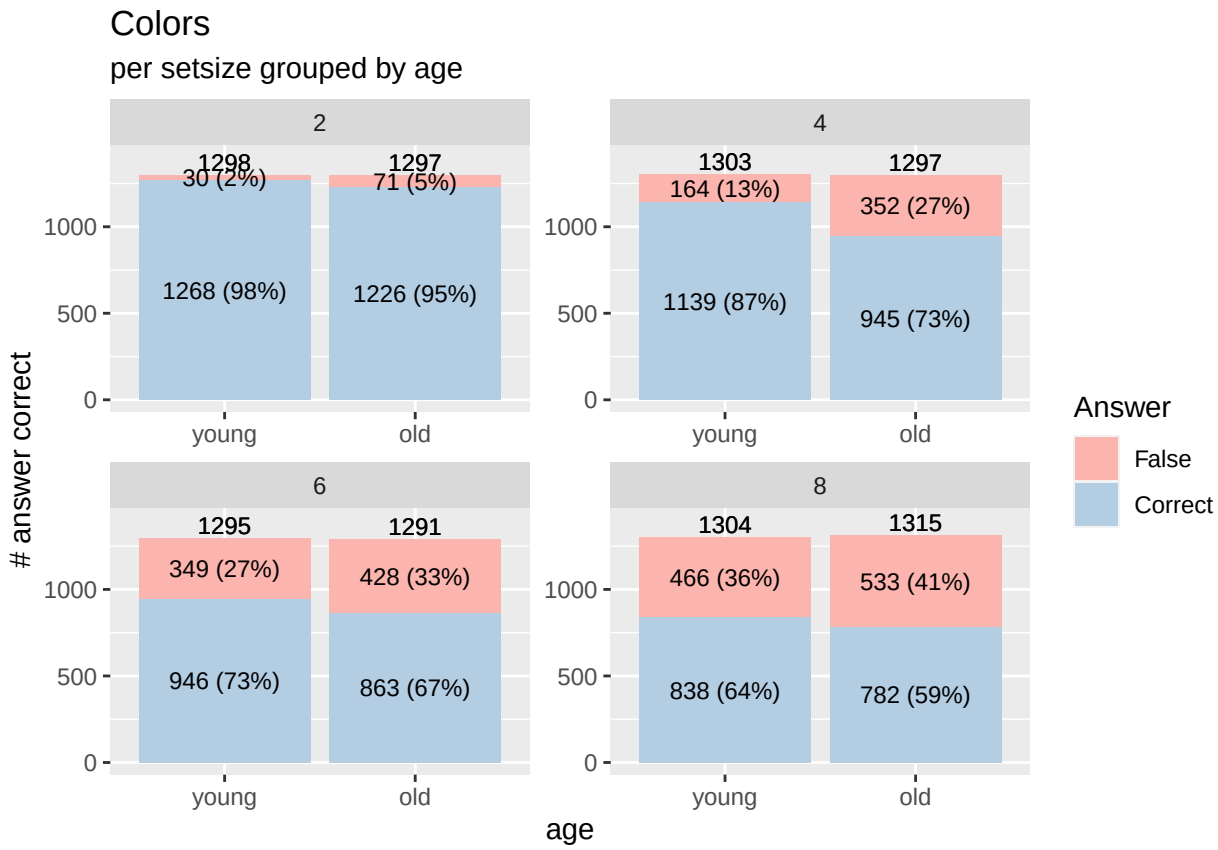
SetSizeTotal	SetSize2	SetSize4	SetSize6	K_Total_mean	K_2_mean	K_6_mean	K_PTotal_mean	K_P2_mean	K_P4_mean
-0.03	-0.01	-0.1	0	-0.31	-0.06	-0.02	-0.48	-0.01	-0.59

Plotte einen Überblick der Gesamtmenge

```
colorList %>% mutate(age=factor(age, levels = c("young","old")),
              answer_correct=factor(answer_correct, levels = c(0,1),labels = c("False","Correct"))) %>%
  group_by(age,condition,answer_correct) %>% summarise(N=n()) %>% ungroup() %>%
  group_by(age,condition) %>%
  mutate(Total=sum(N),Percent=N/Total,
         Lab=paste0(N,' (',paste0(round(100*Percent,0),'%'),')')) -> SumsColors
```

```
## `summarise()` regrouping output by 'age', 'condition' (override with `.groups` argument)
```

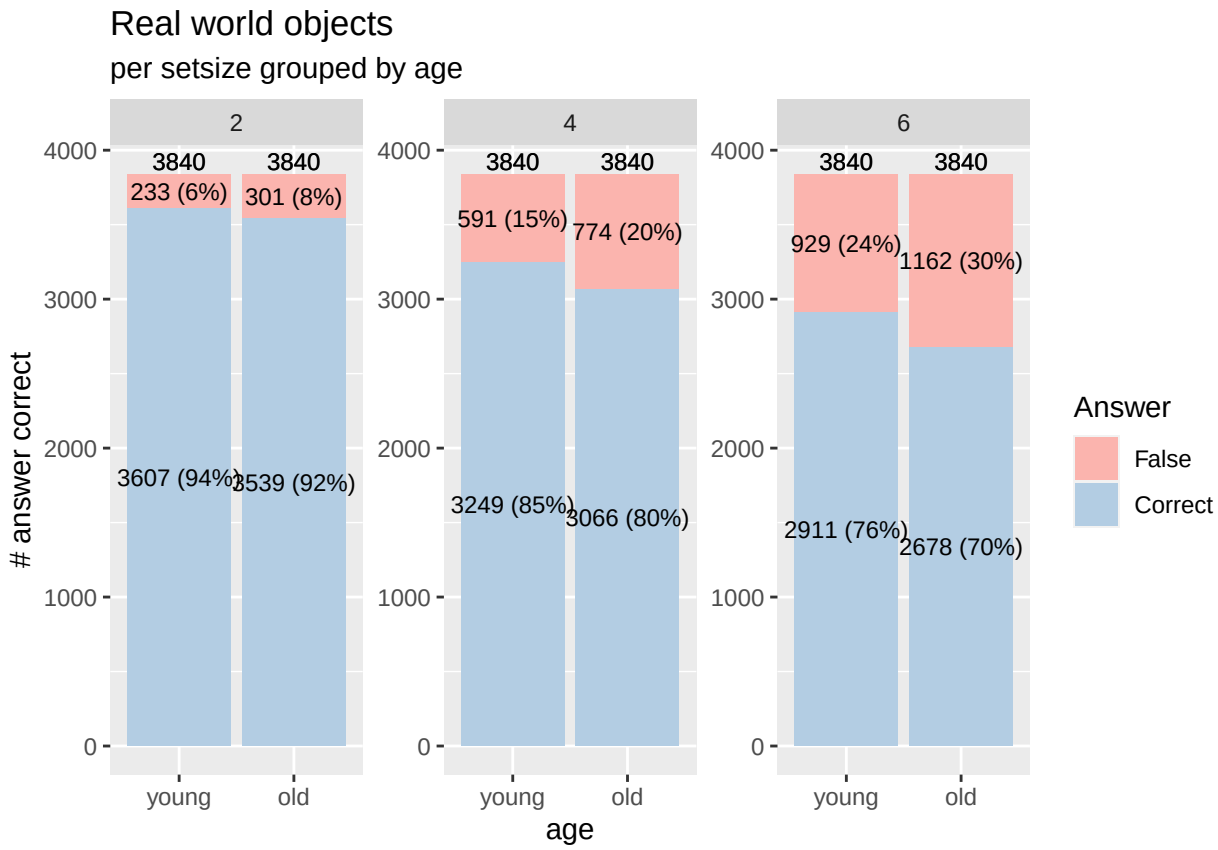
```
#Plot
ggplot(SumsColors,aes(x=age,y=N,fill=answer_correct))+
  geom_bar(stat='identity',position = position_stack())+
  facet_wrap(.~condition,scales = 'free')+
  geom_text(aes(label=Lab),position = position_stack(vjust = .5),size=3)+
  geom_text(aes(y=Total,label=Total),vjust=-0.25,size=3)+
  labs(title="Colors",subtitle = "per setsize grouped by age", x="age", y="# answer correct", fill="Ans
  ylim(0, 1400)
```




```
objectList %>% mutate(age=factor(age, levels = c("young","old")),
             answer_correct=factor(answer_correct, levels = c(0,1),labels = c("False","Correct"))) %>%
  group_by(age,setsize,answer_correct) %>% summarise(N=n()) %>% ungroup() %>%
  group_by(age,setsize) %>%
  mutate(Total=sum(N),Percent=N/Total,
         Lab=paste0(N,' (',paste0(round(100*Percent,0),'%'),')')) -> SumsObjects
```

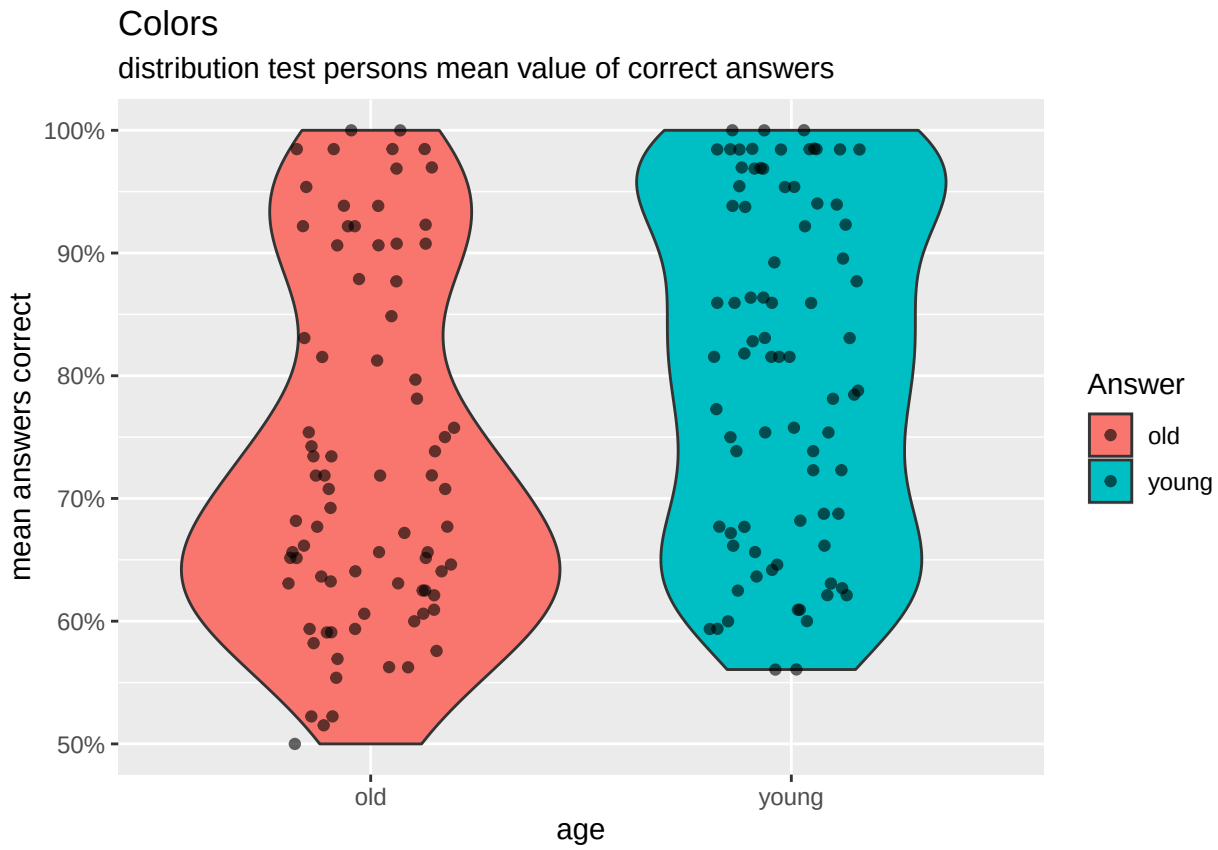
```
## `summarise()` regrouping output by 'age', 'setsize' (override with `.groups` argument)
```

```
#Plot
ggplot(SumsObjects,aes(x=age,y=N,fill=answer_correct))+
  geom_bar(stat='identity',position = position_stack())+
  facet_wrap(.~setsize,scales = 'free')+
  geom_text(aes(label=Lab),position = position_stack(vjust = .5),size=3)+
  geom_text(aes(y=Total,label=Total),vjust=-0.25,size=3)+
  labs(title="Real world objects",subtitle = "per setsize grouped by age", x="age", y="# answer correct
```

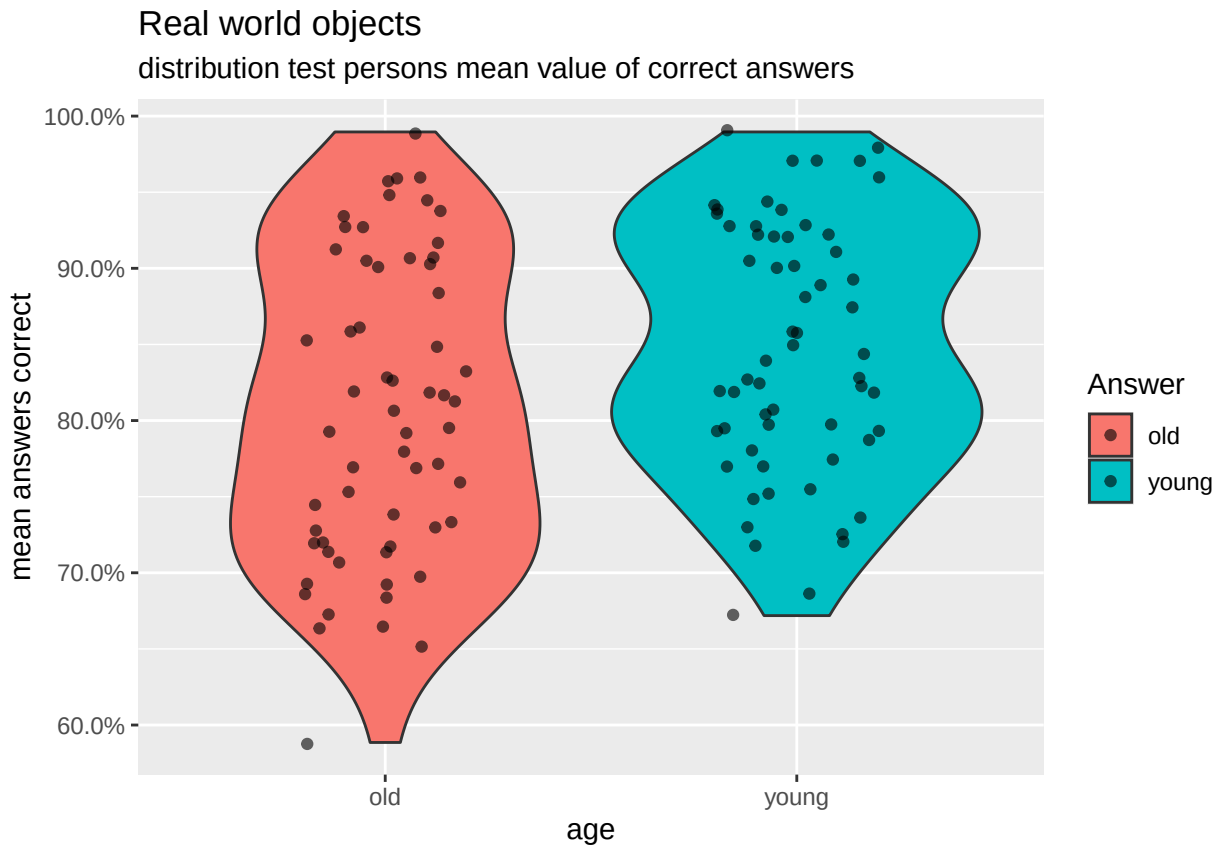


Plotte die Streuung von Colors und real objects

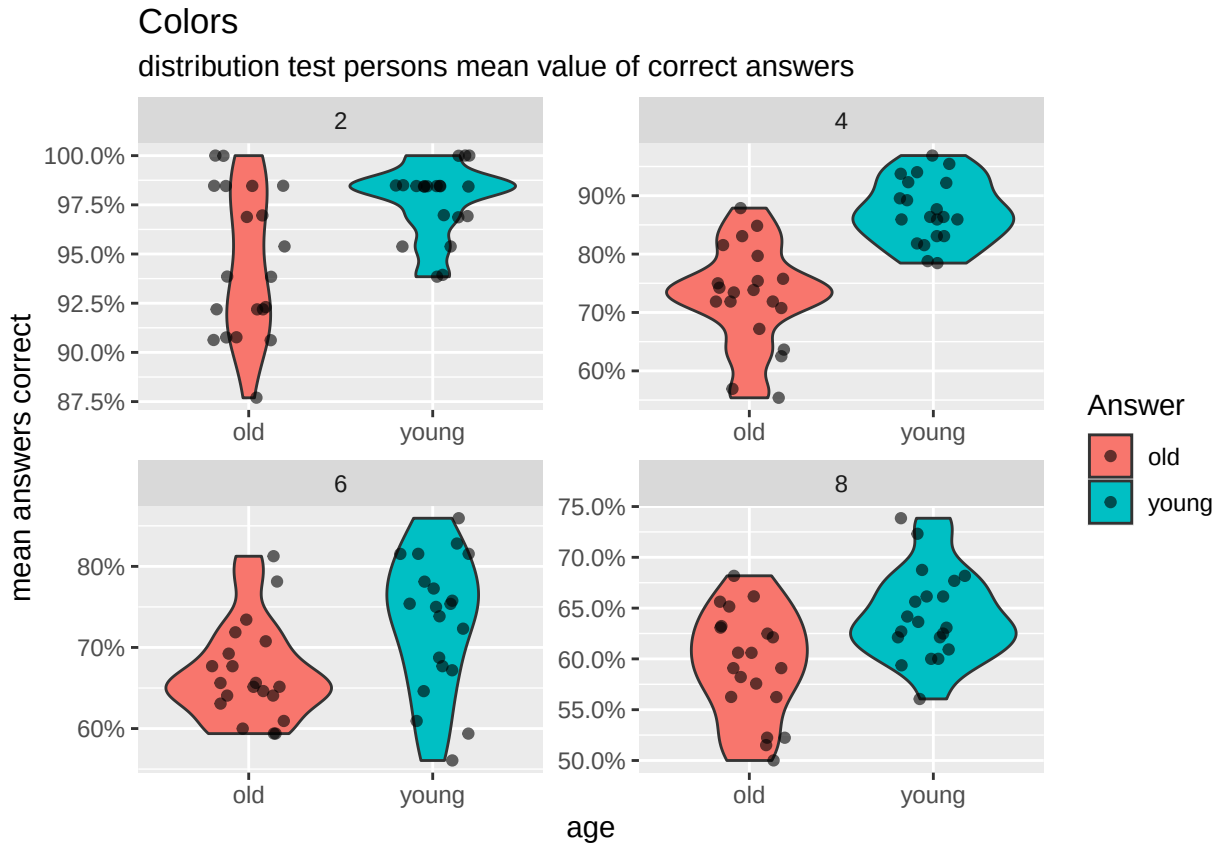
```
colorListResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age)) + geom_violin() + geom_
```



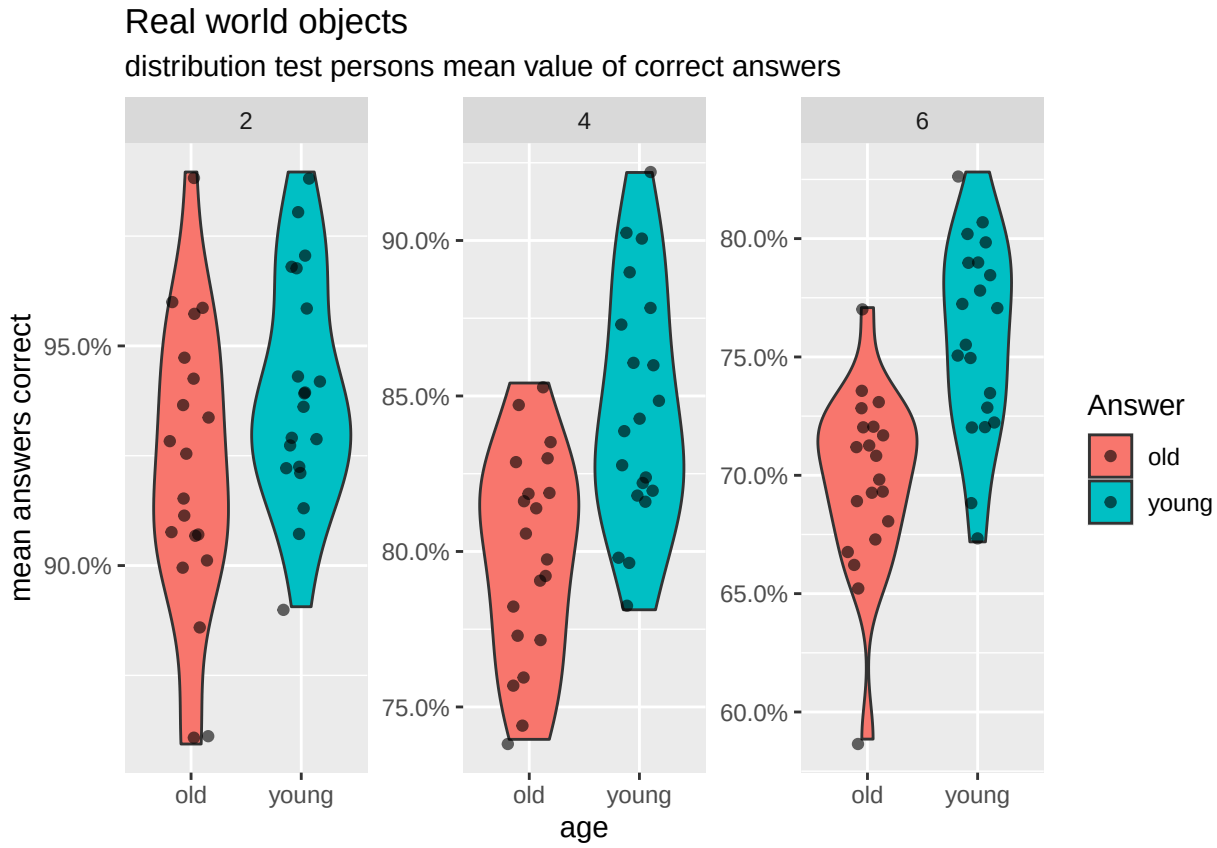

```
objectResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age)) + geom_violin() + geom_jit
```




```
colorListResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age))+ geom_violin() + geom_j
```

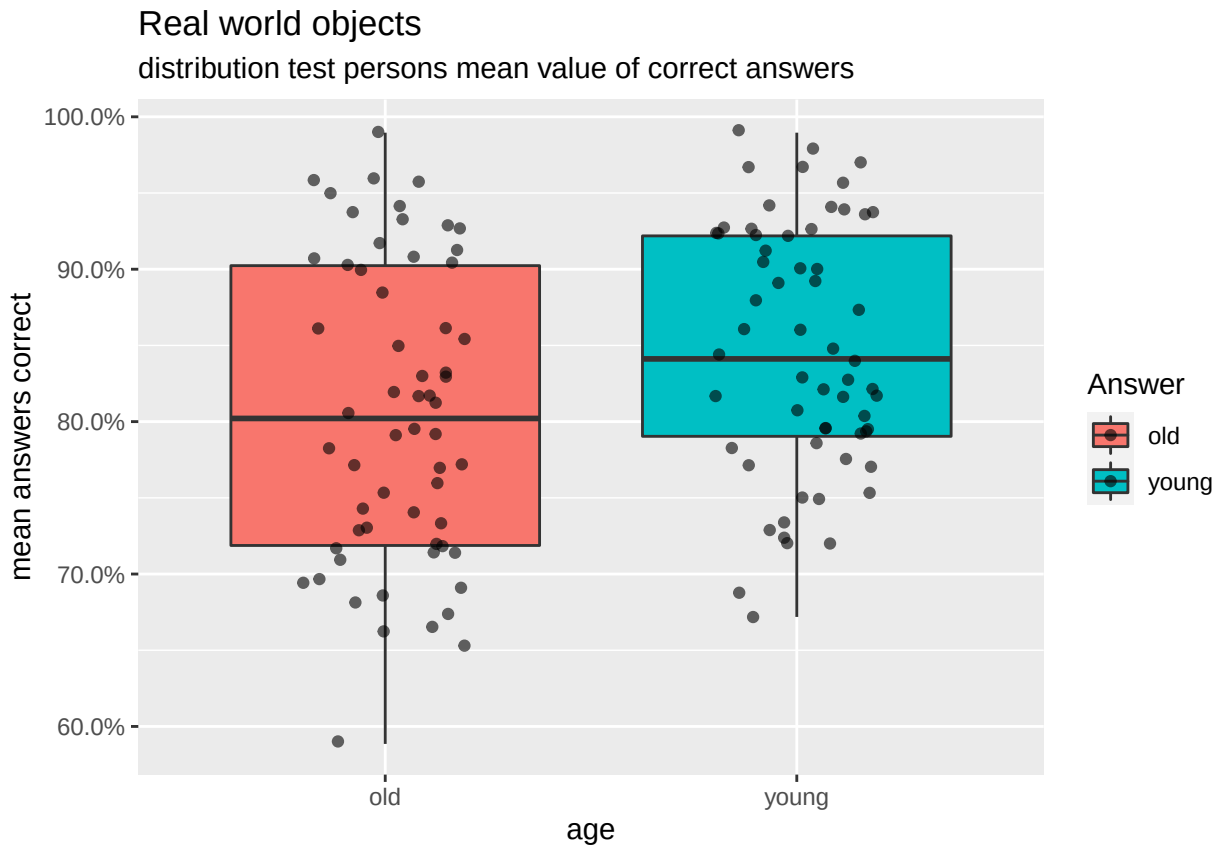



```
objectResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age))+ geom_violin() + geom_jitt
```

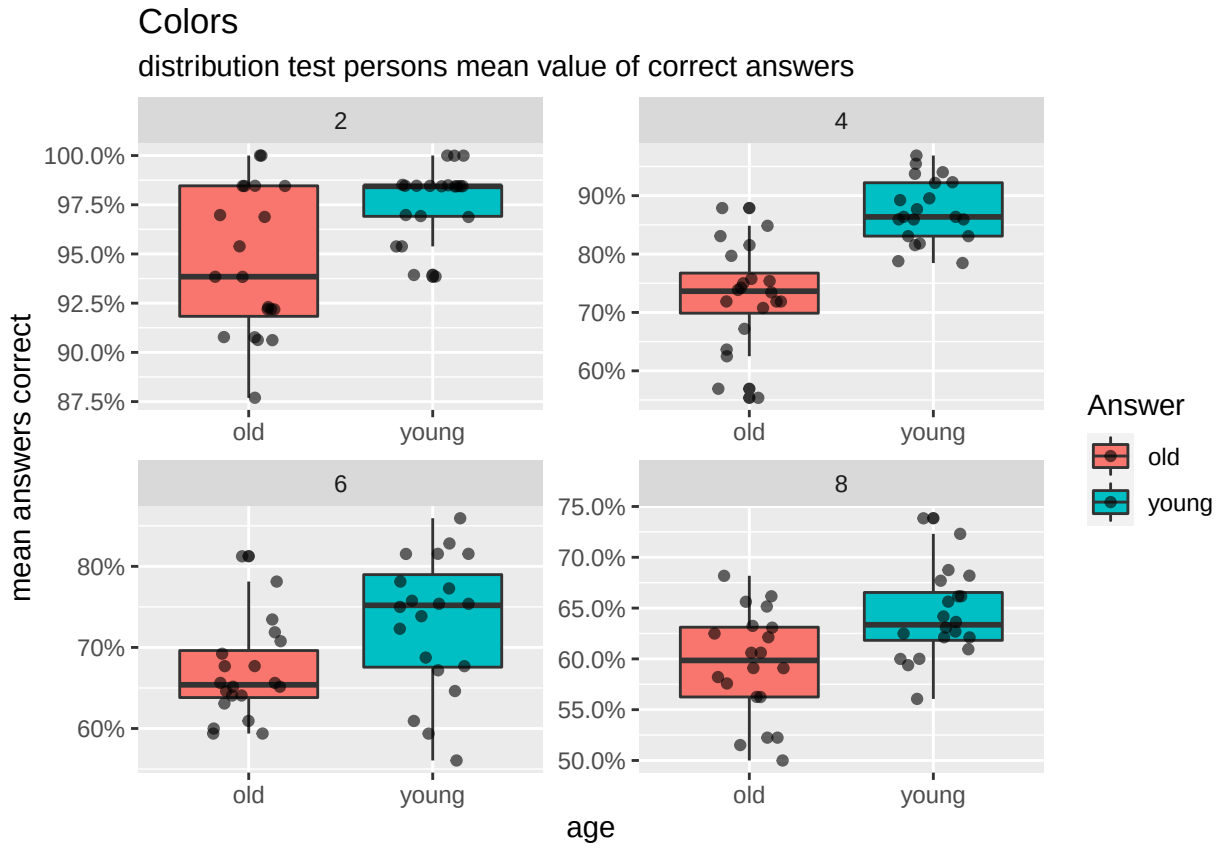



```
colorListResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age)) + geom_boxplot() + geom_
```

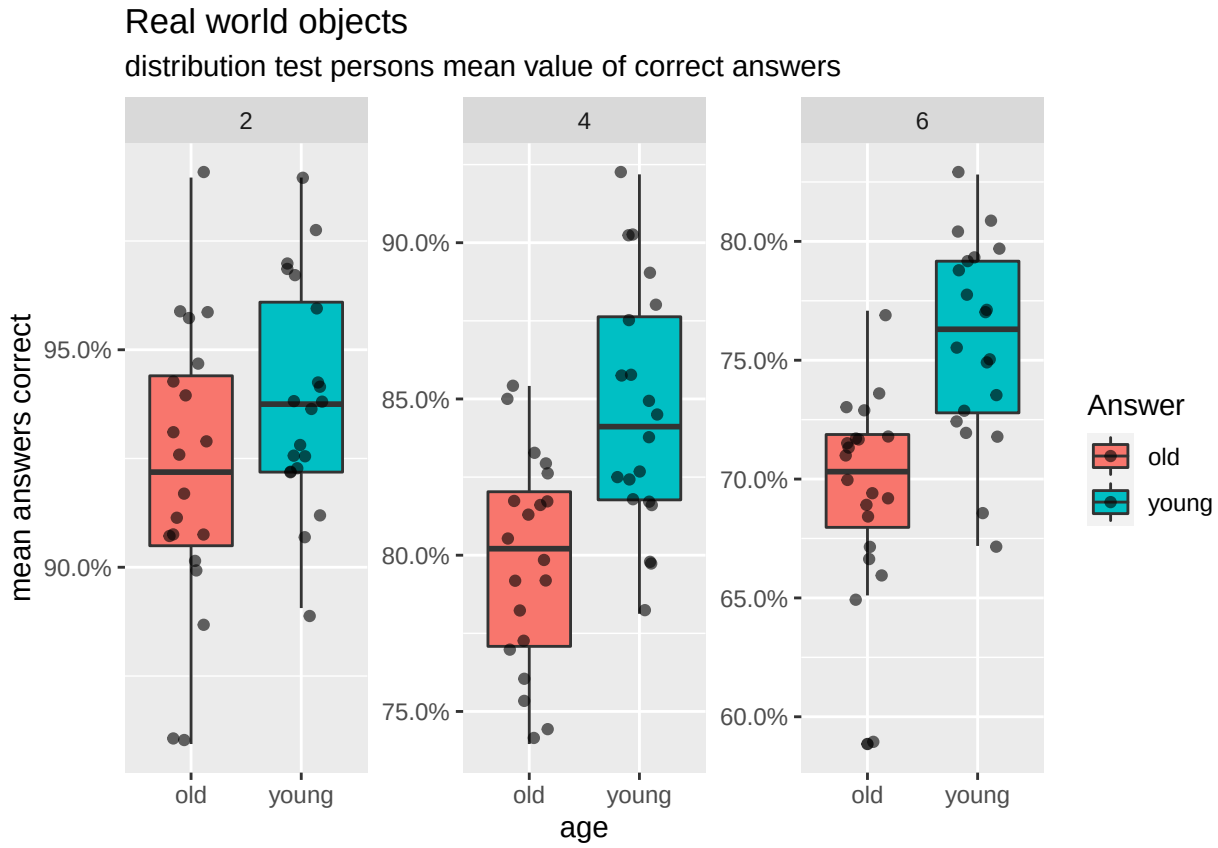

```
objectResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age)) + geom_boxplot() + geom_ji
```




```
colorListResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age))+ geom_boxplot() + geom_
```

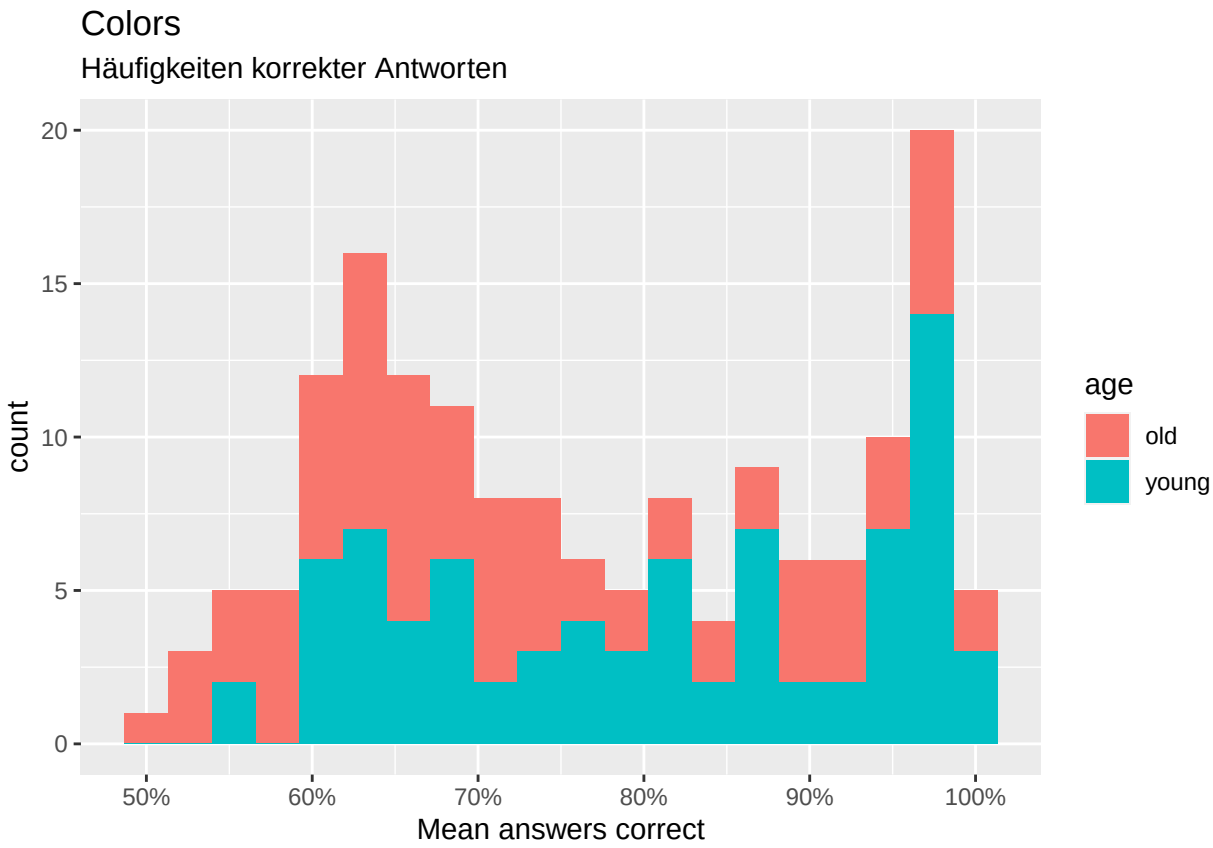



```
objectResult %>%  ggplot(mapping=aes(x=age, y=mean_answer_correct,fill=age))+ geom_boxplot() + geom_jit
```



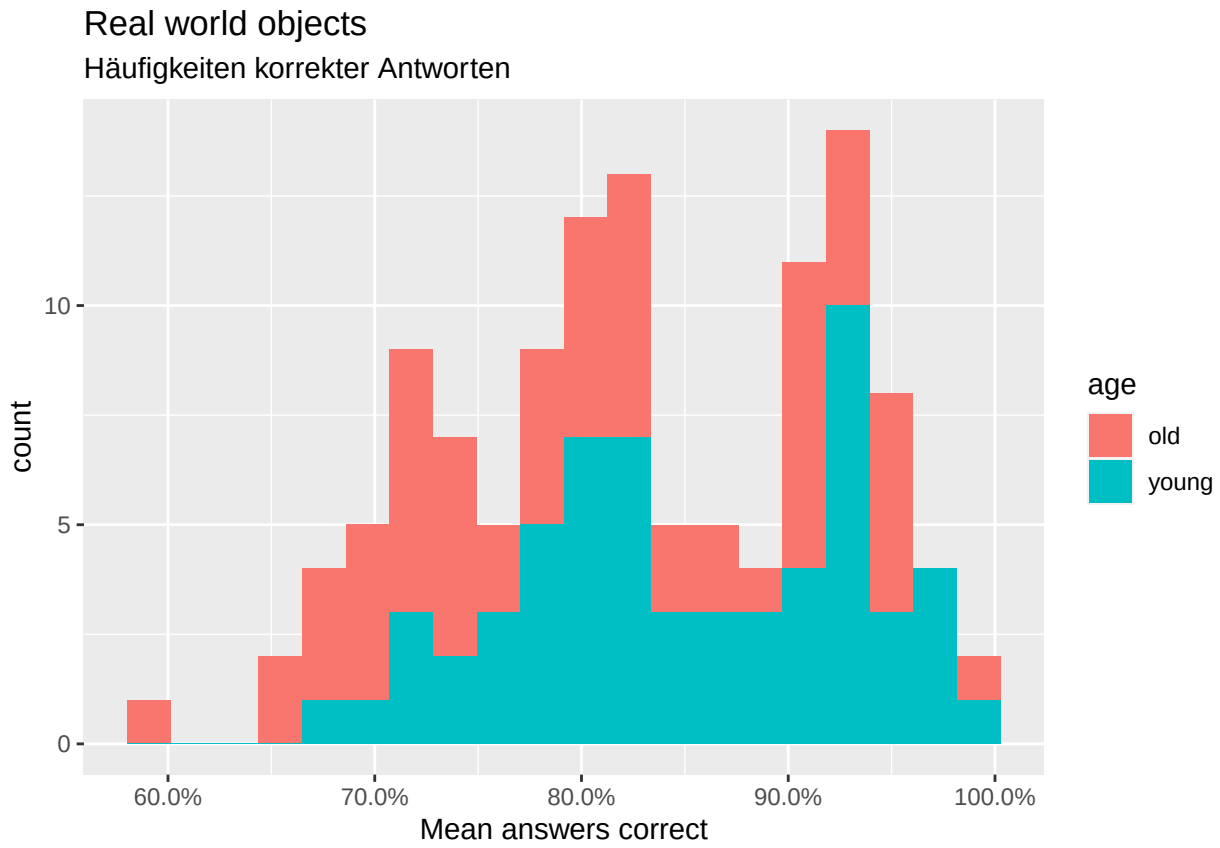
Häufigkeitsverteilung

```
colorListResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_histogram(bins = 20)+la
```

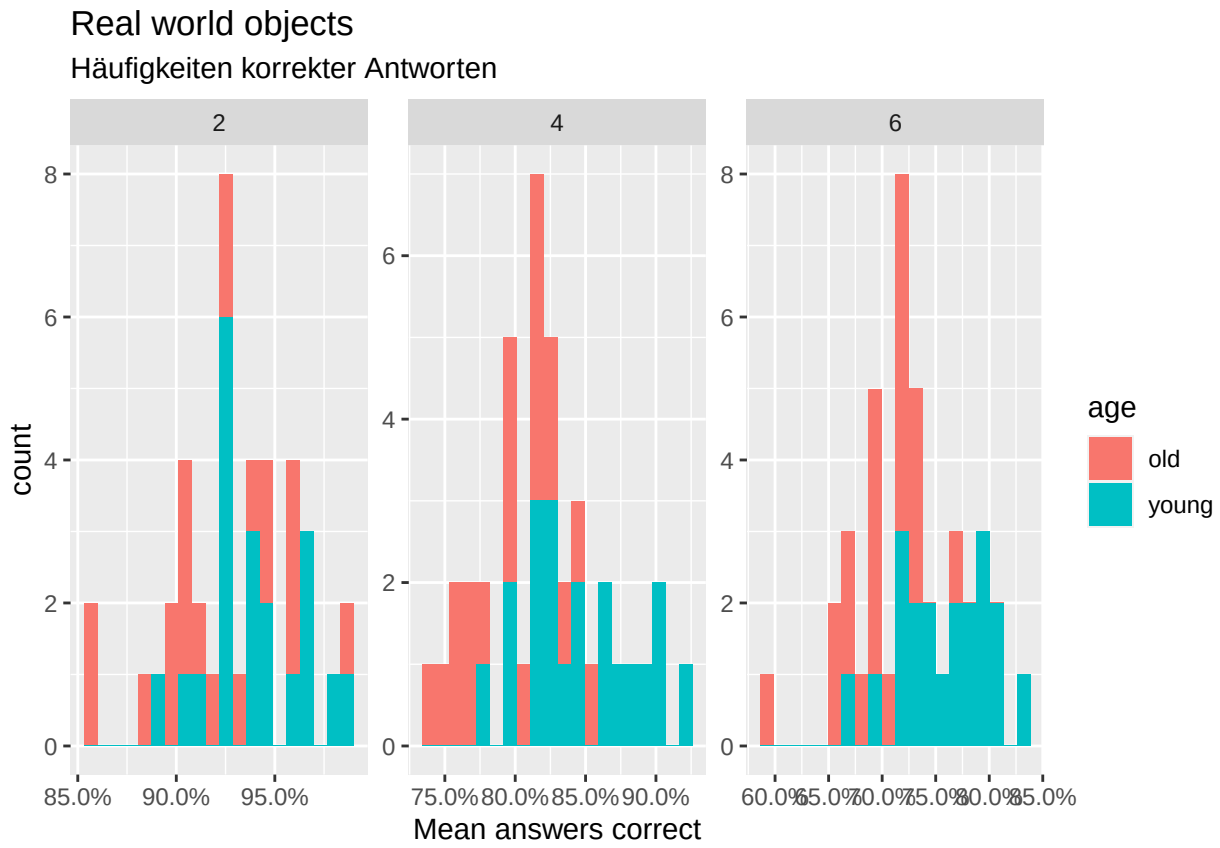



```
colorListResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_histogram(bins=20)+face
```

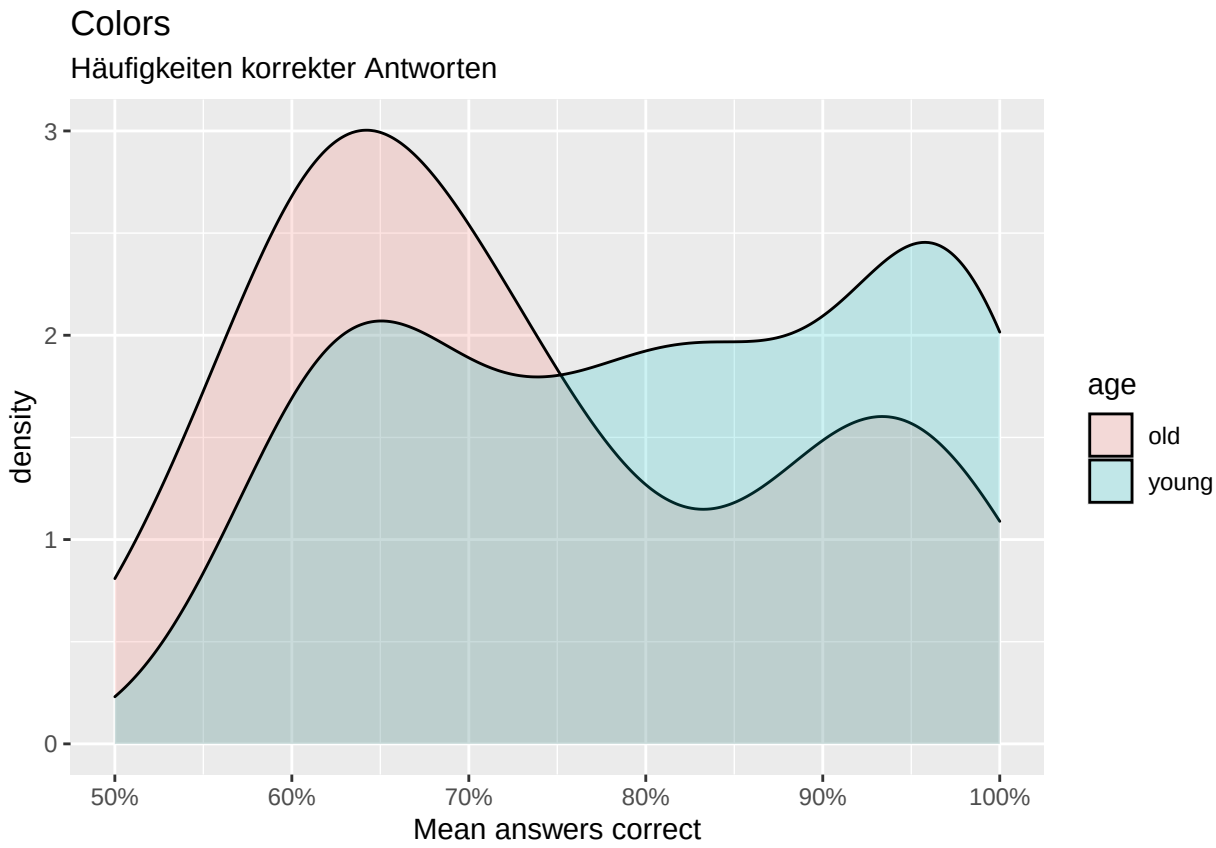

```
objectResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_histogram(bins=20)+labs(ti
```



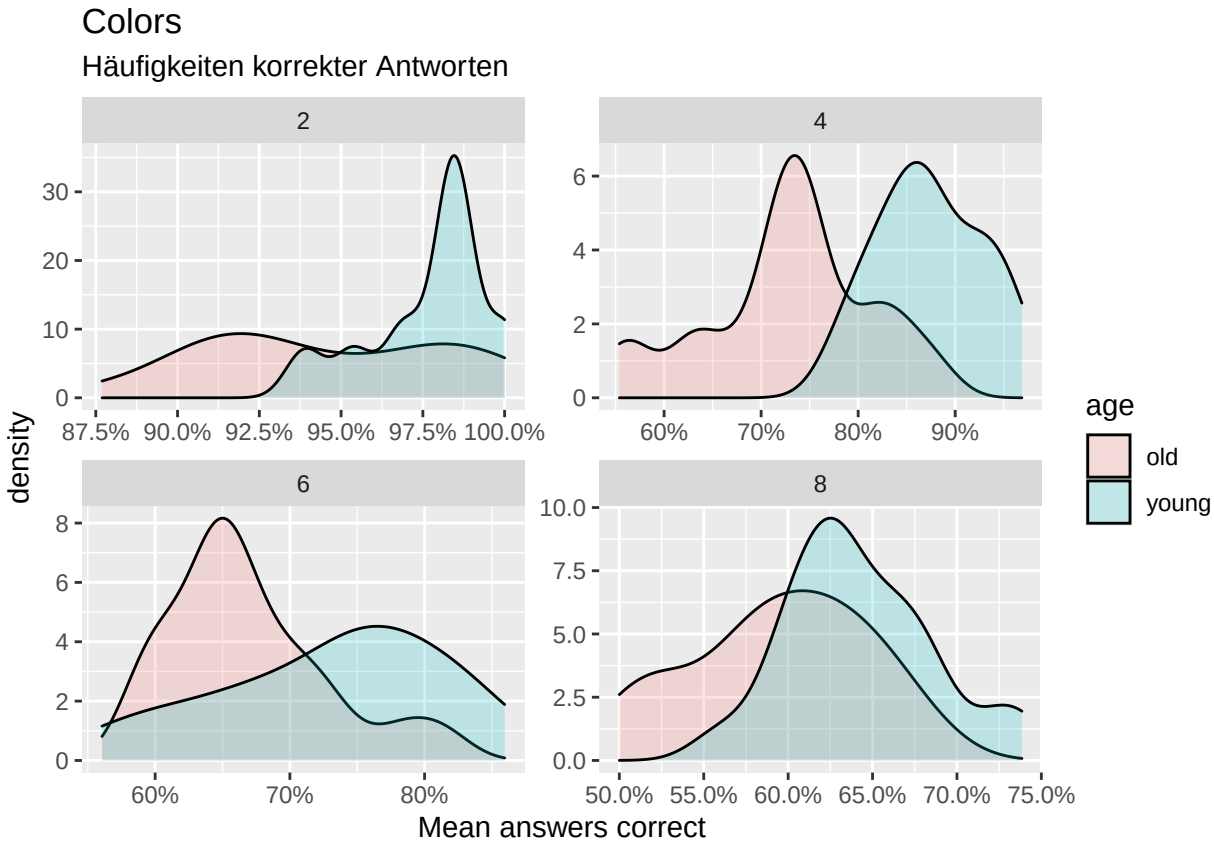

```
objectResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_histogram(bins=20)+facet_wr
```



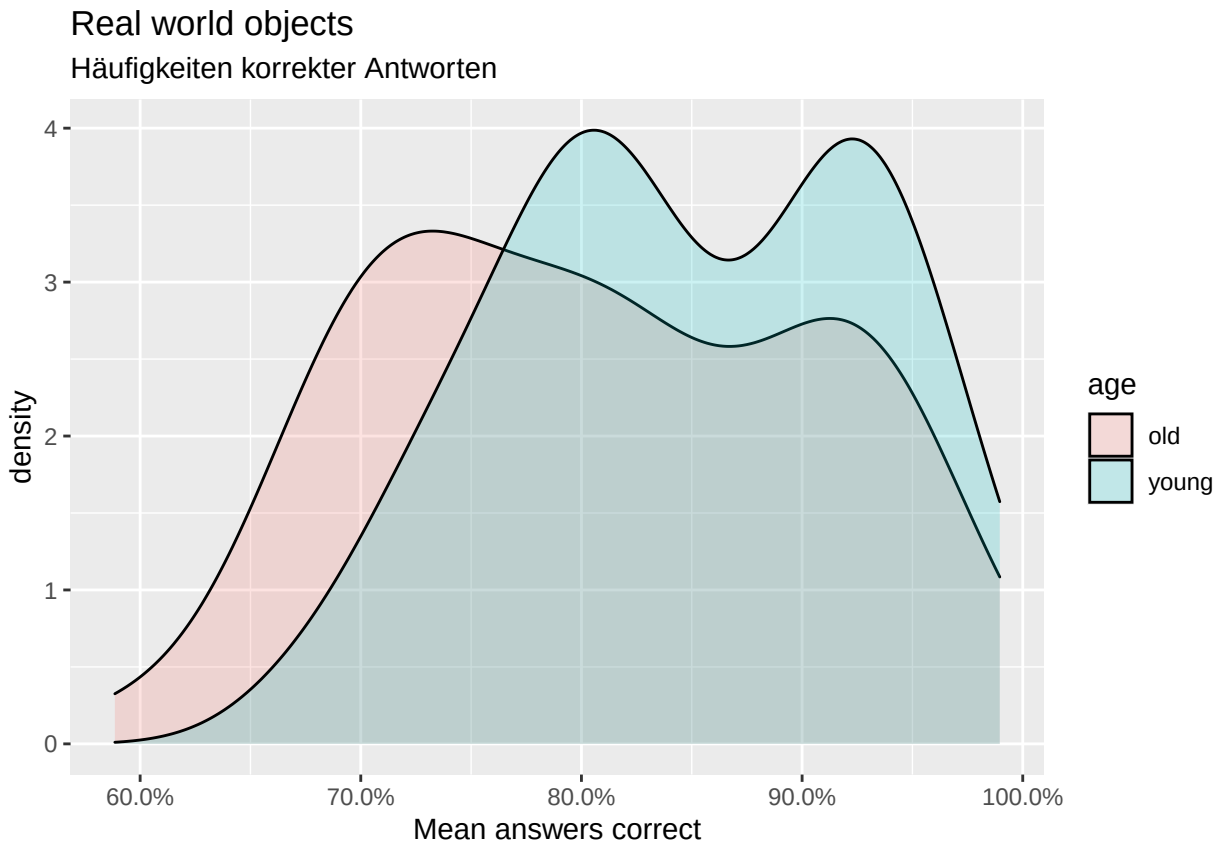

```
colorListResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_density (alpha=0.2)+lab
```



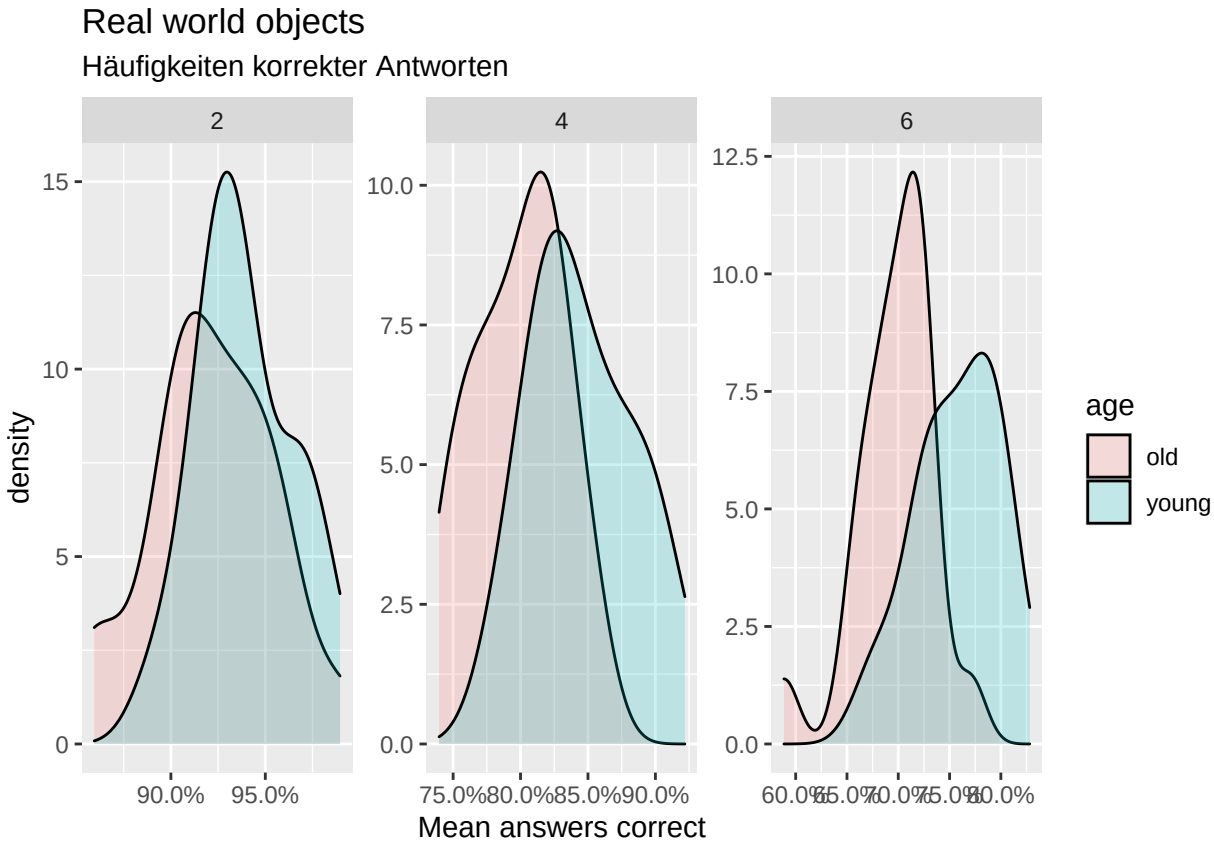

```
colorListResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_density( alpha=0.2)+fac
```




```
objectResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_density(alpha=0.2)+labs(ti
```

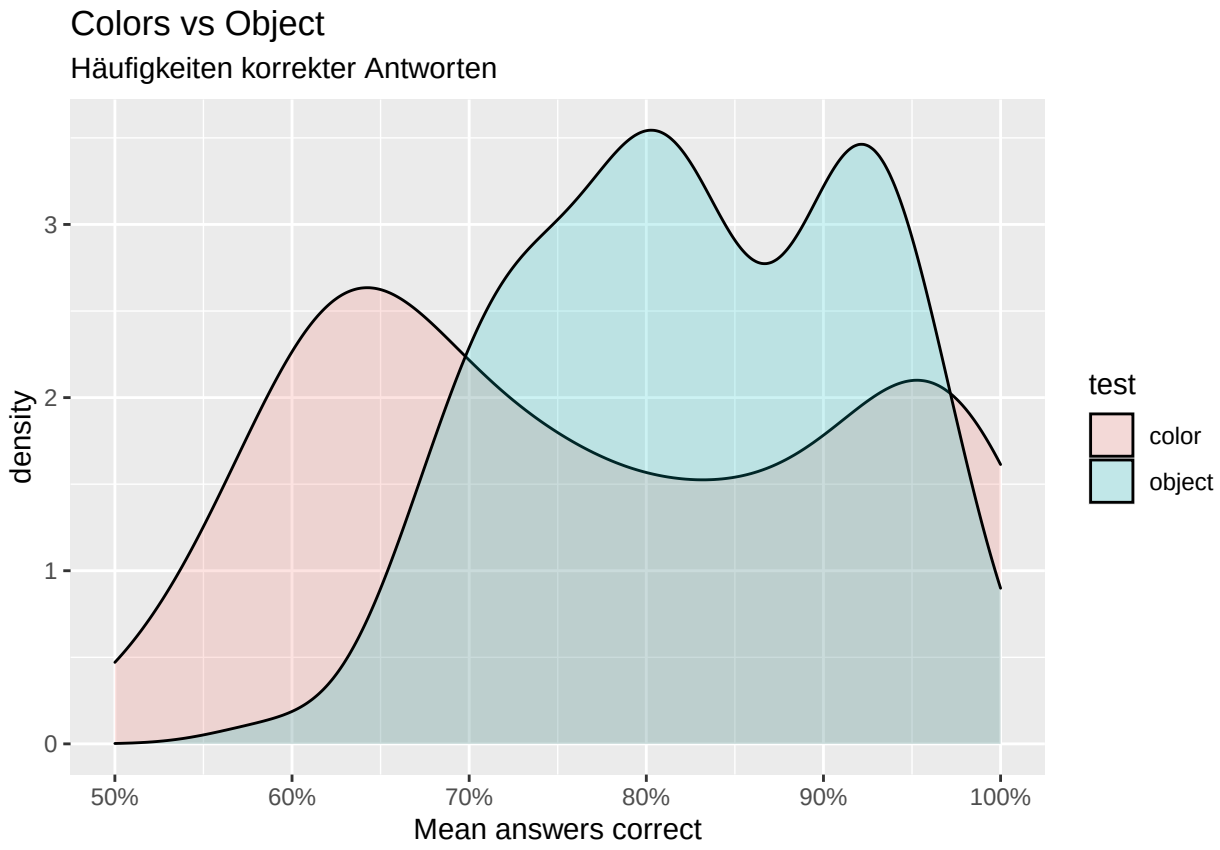



```
objectResult %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=age))+ geom_density(alpha=0.2)+facet_wr
```

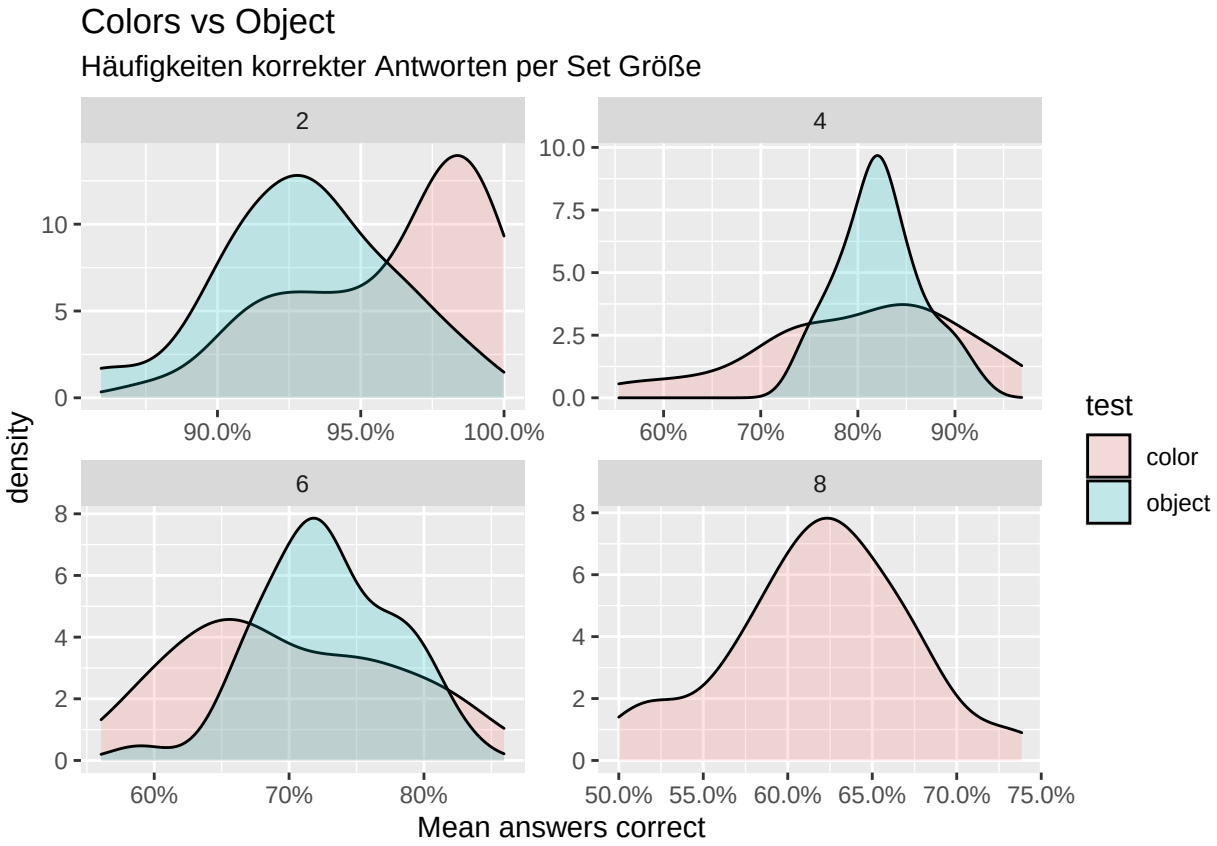


Vergleich Color vs. Objects gleiche Altersgruppen

```
joinedTestList %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=test))+ geom_density (alpha=0.2)+labs
```



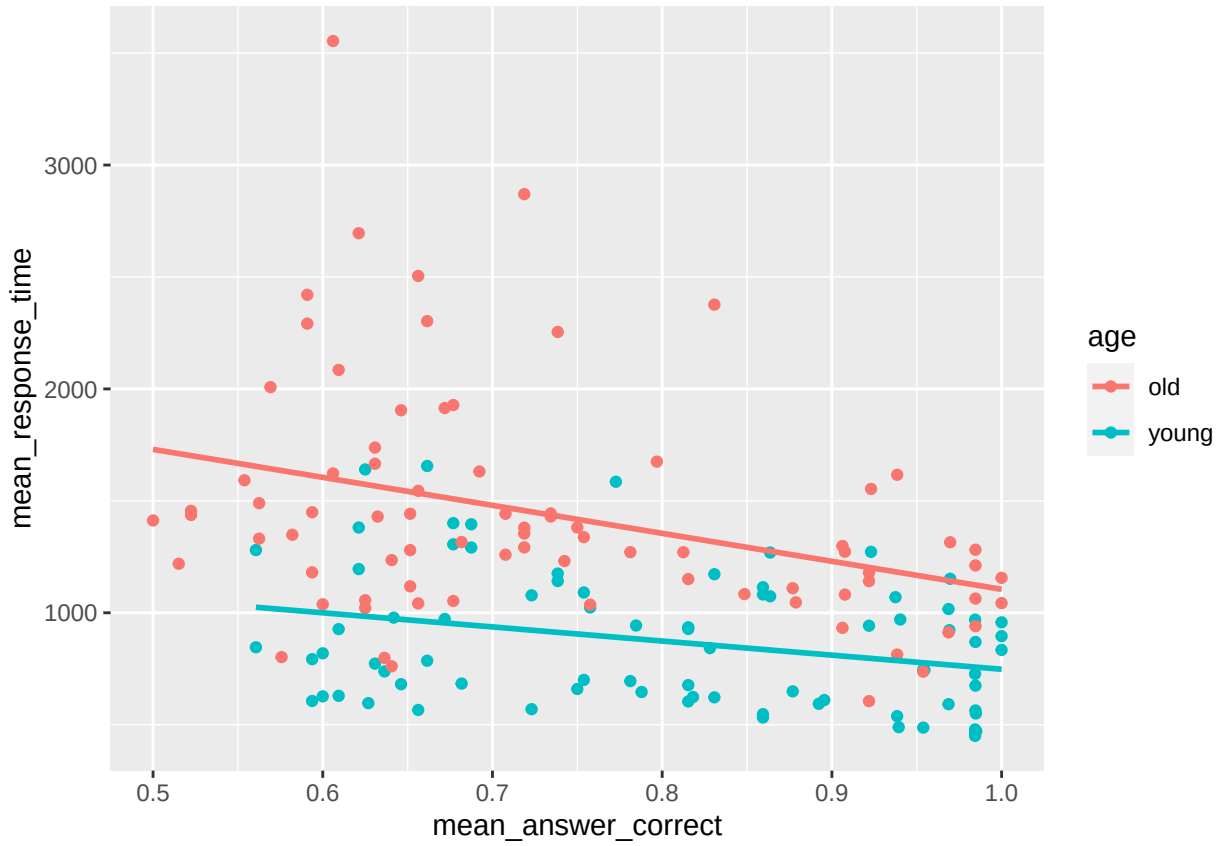

```
joinedTestList %>%  ggplot(mapping=aes(x=mean_answer_correct, fill=test))+ geom_density (alpha=0.2)+fac
```



Korrelation

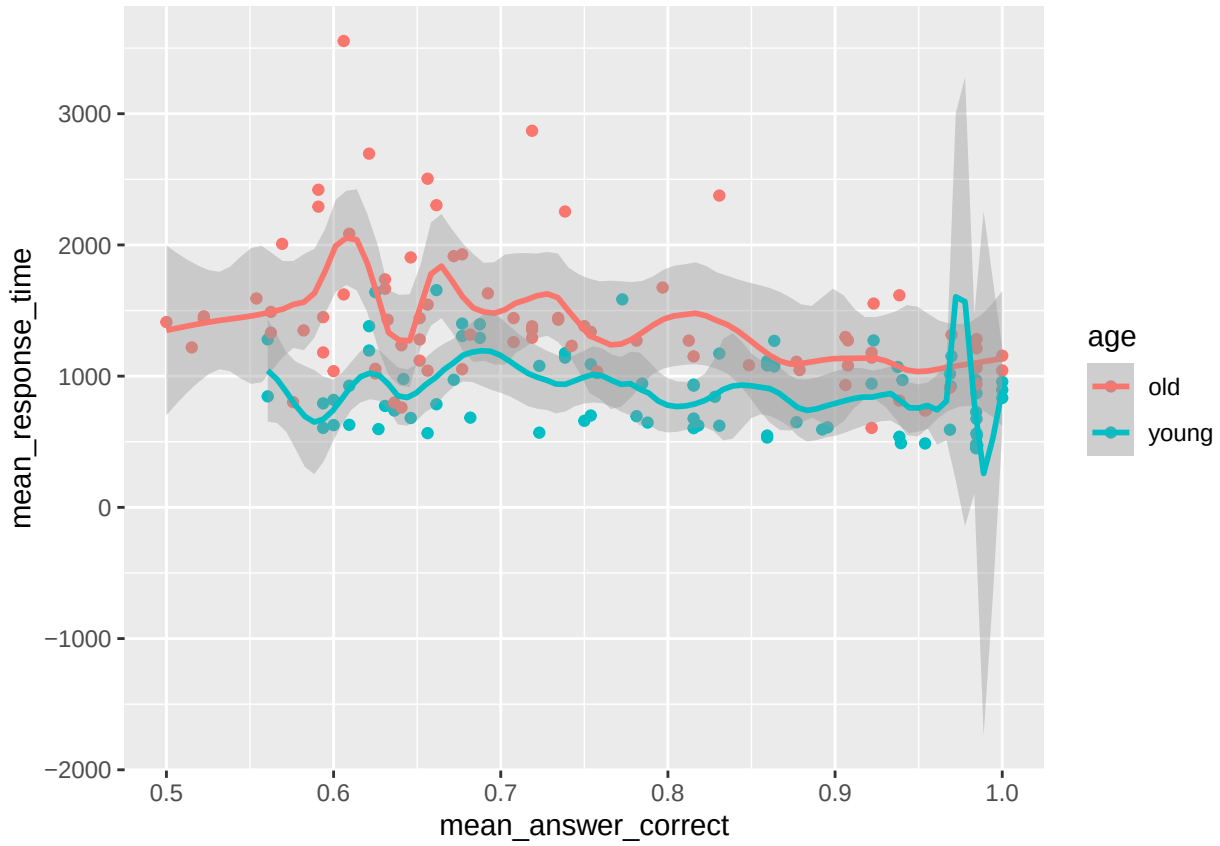
```
colorListResult %>% ggplot(aes(x=mean_answer_correct, y=mean_response_time,color = age))+geom_point()+g
```

```
## `geom_smooth()` using formula 'y ~ x'
```



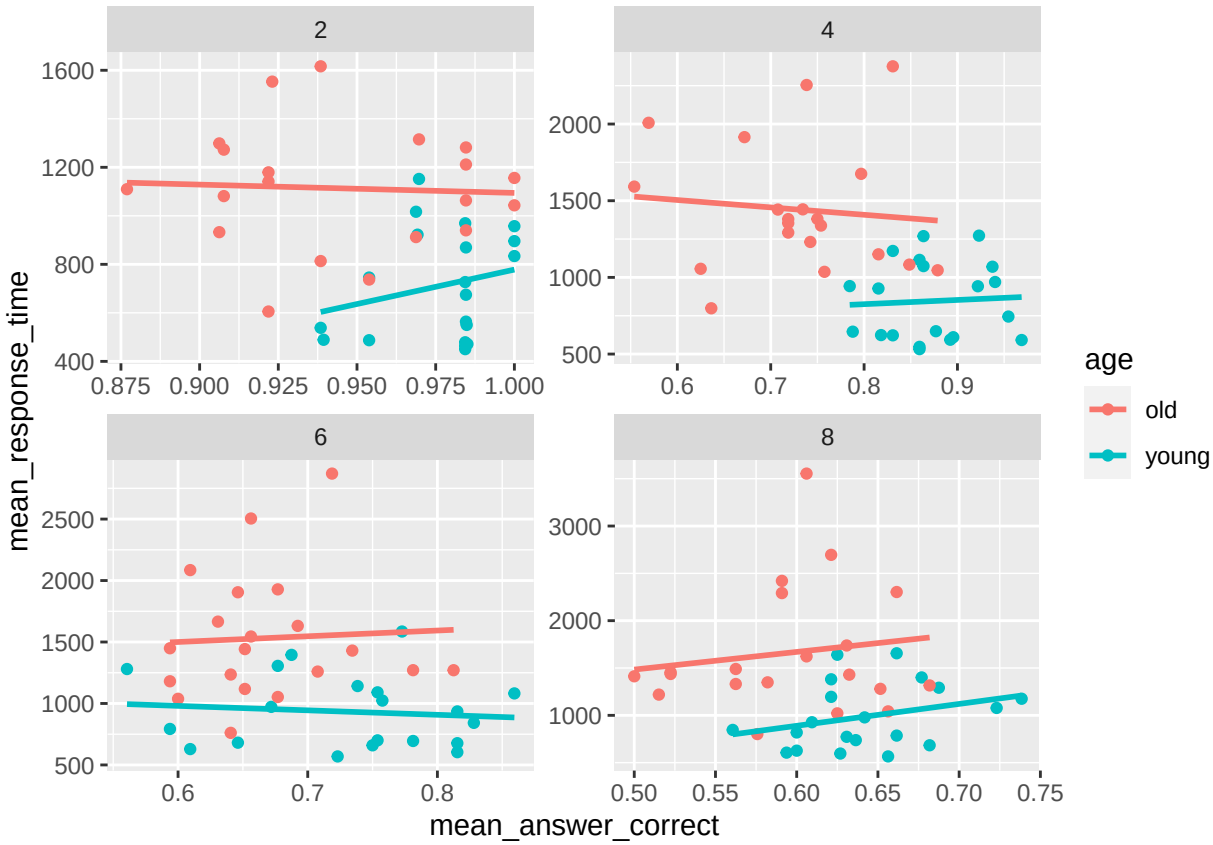

```
colorListResult %>% ggplot(aes(x=mean_answer_correct, y=mean_response_time,color = age))+geom_point()+g
```

```
## `geom_smooth()` using formula 'y ~ x'
```



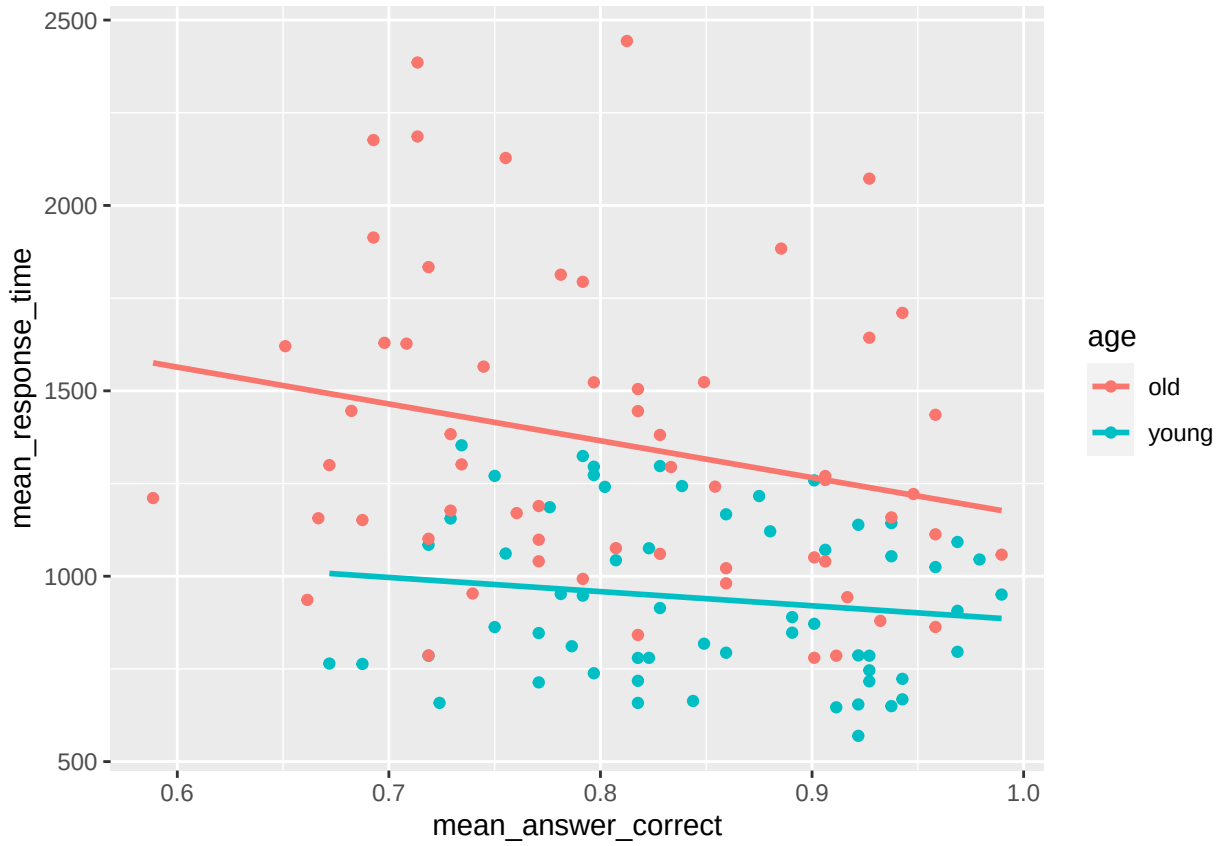

```
colorListResult %>% ggplot(aes(x=mean_answer_correct, y=mean_response_time,color = age))+geom_point()+g
```

```
## `geom_smooth()` using formula 'y ~ x'
```



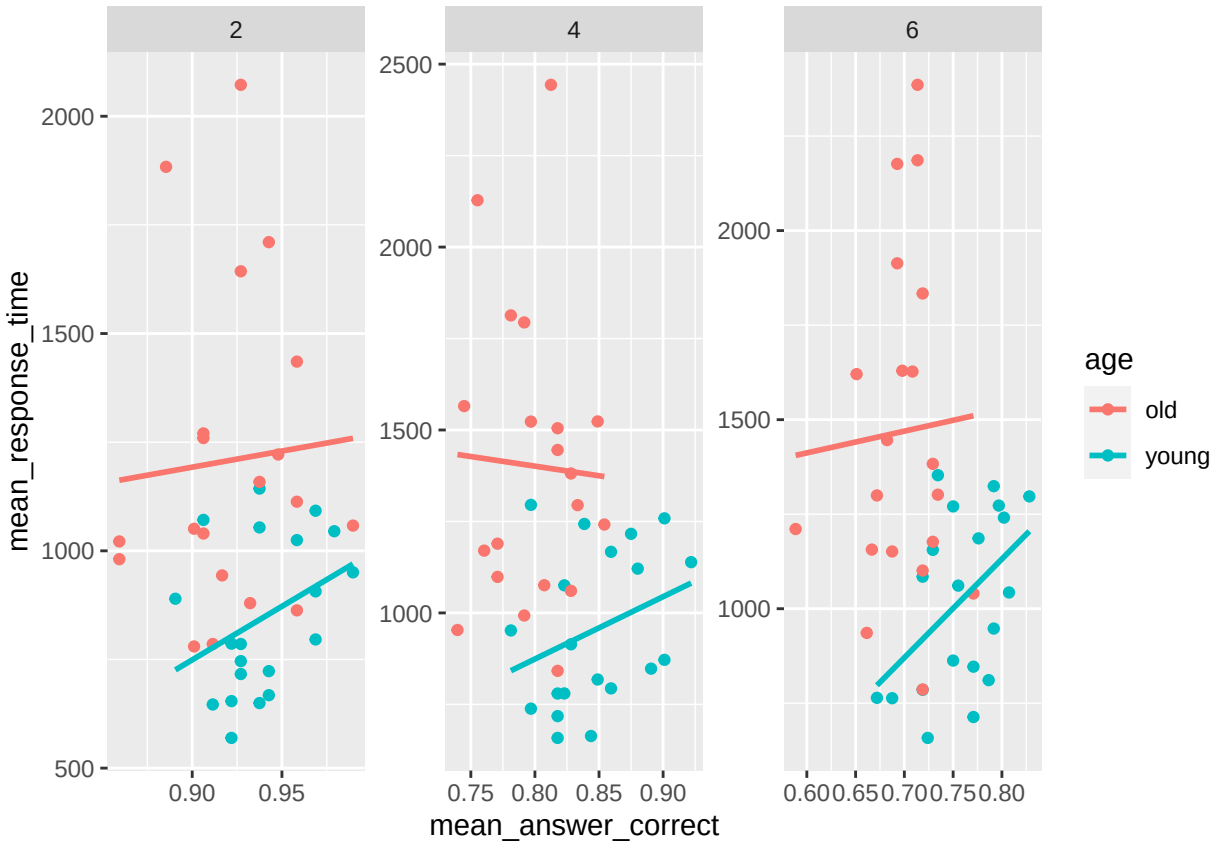

```
objectResult %>% ggplot(aes(x=mean_answer_correct, y=mean_response_time,color = age))+geom_point()+geom_
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
objectResult %>% ggplot(aes(x=mean_answer_correct, y=mean_response_time,color = age))+geom_point()+geom_
```

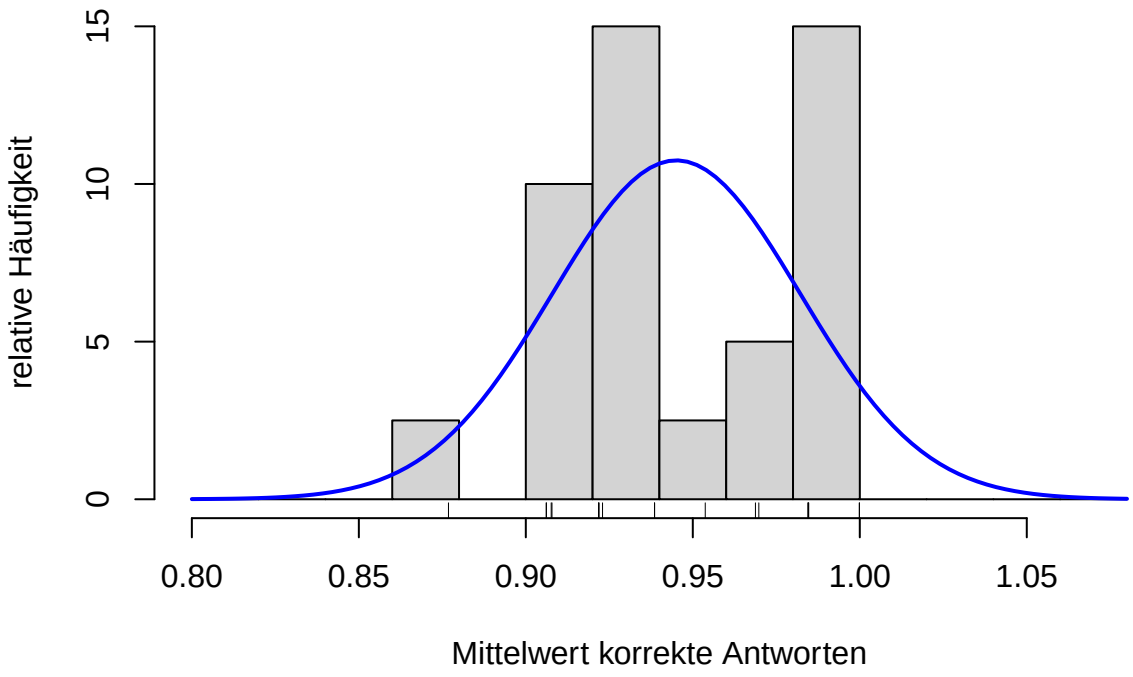
```
## `geom_smooth()` using formula 'y ~ x'
```



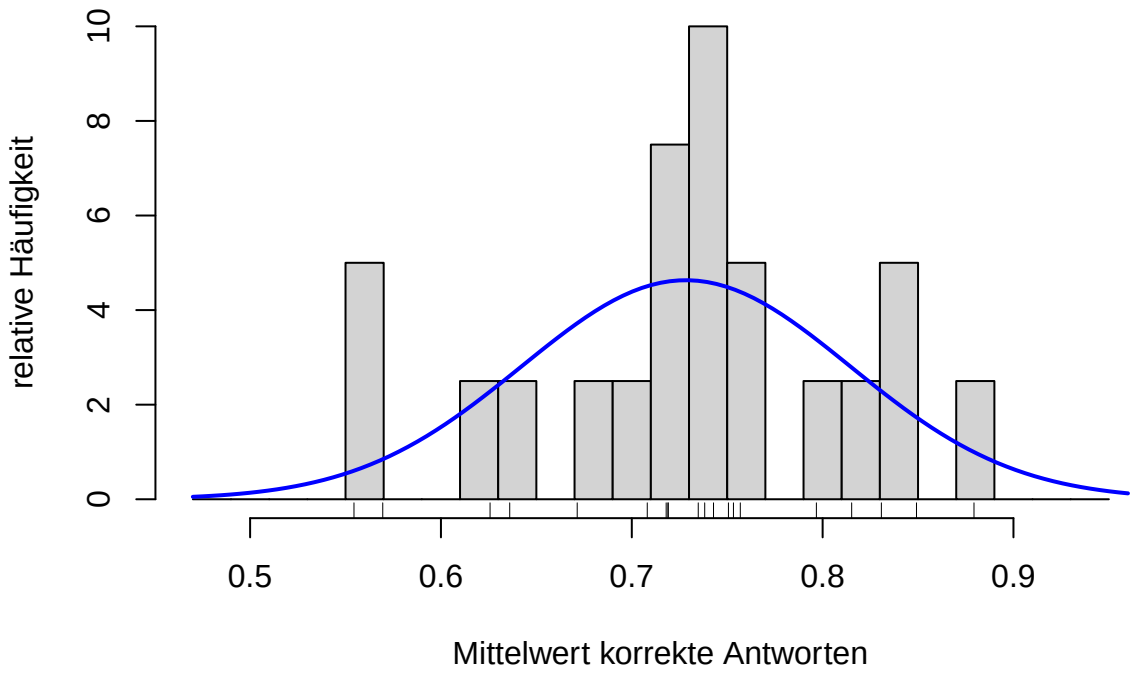
ob-
jectResult %>% filter(age==“young”,setsize==6) %>% ungroup %>% select(mean_answer_correct) ->
testIt shapiro.test(testIt$mean_answer_correct)

Normalverteilung:

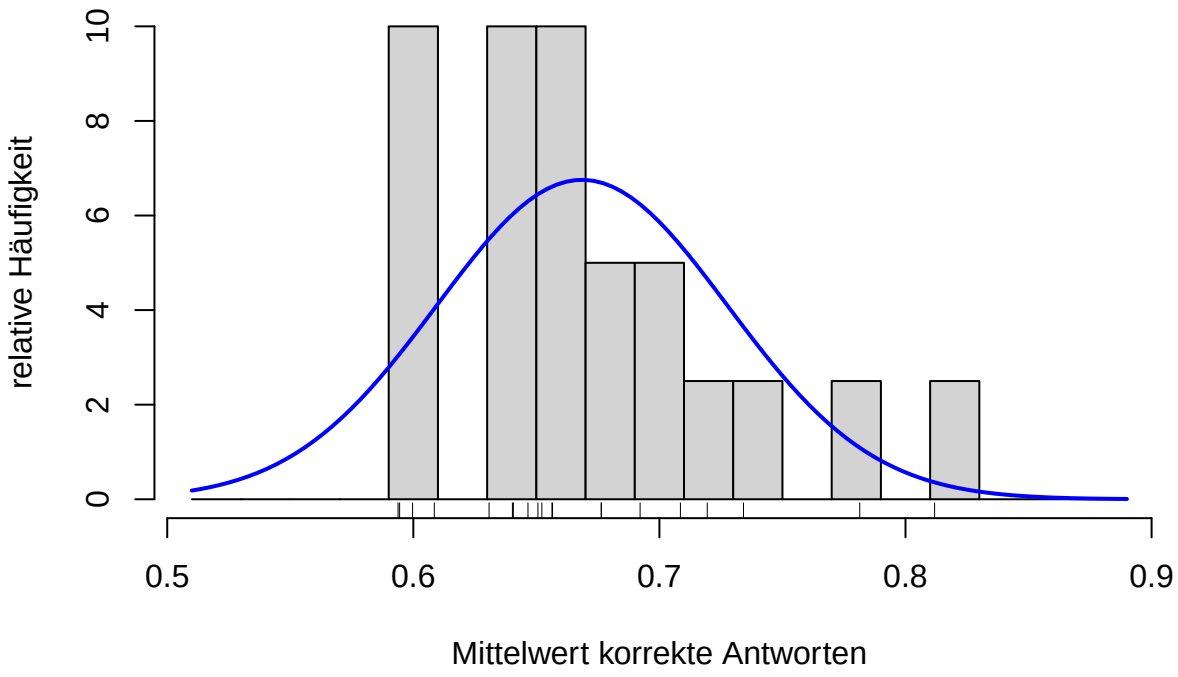
```
colorListResult %>% filter(condition==2,age=="old") -> df_means_o2
fromTo <- round(range(df_means_o2$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_o2$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_o2$mean_answer_correct))
curve(dnorm(x,mean(df_means_o2$mean_answer_correct),sd(df_means_o2$mean_answer_correct)),lwd=2,col="blu
```



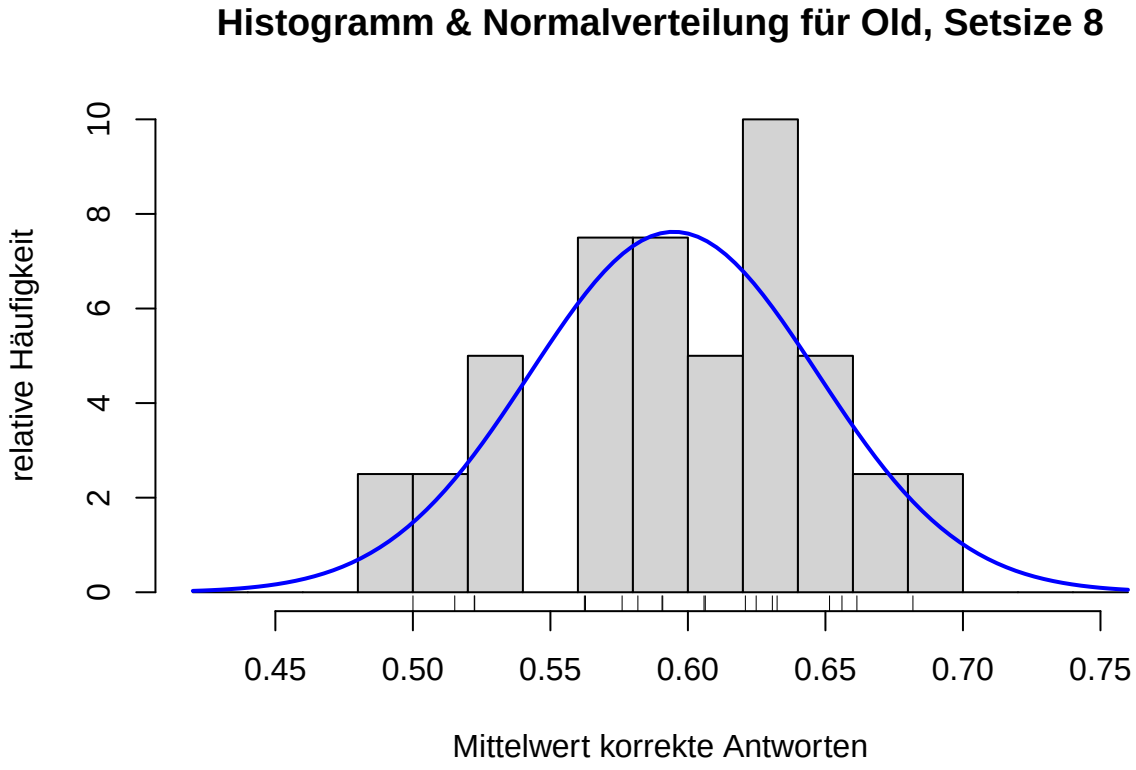

```
colorListResult %>% filter(condition==4,age=="old") -> df_means_o4
fromTo <- round(range(df_means_o4$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_o4$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_o4$mean_answer_correct))
curve(dnorm(x,mean(df_means_o4$mean_answer_correct),sd(df_means_o4$mean_answer_correct)),lwd=2,col="blu
```



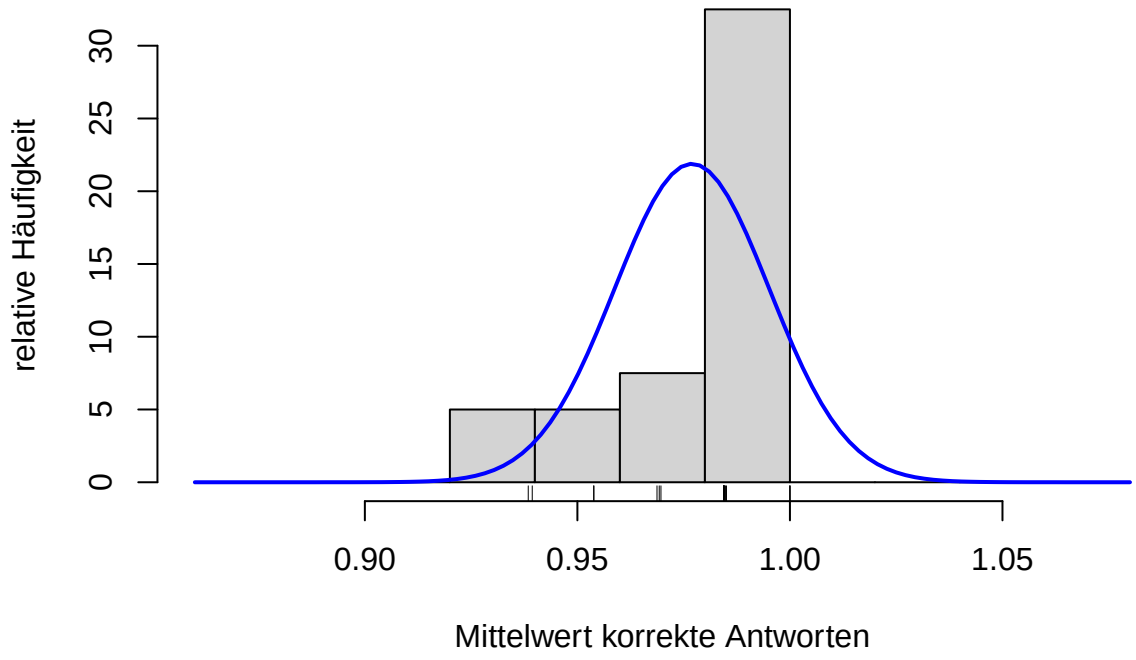

```
colorListResult %>% filter(condition==6,age=="old") -> df_means_o6
fromTo <- round(range(df_means_o6$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_o6$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_o6$mean_answer_correct))
curve(dnorm(x,mean(df_means_o6$mean_answer_correct),sd(df_means_o6$mean_answer_correct)),lwd=2,col="blu
```



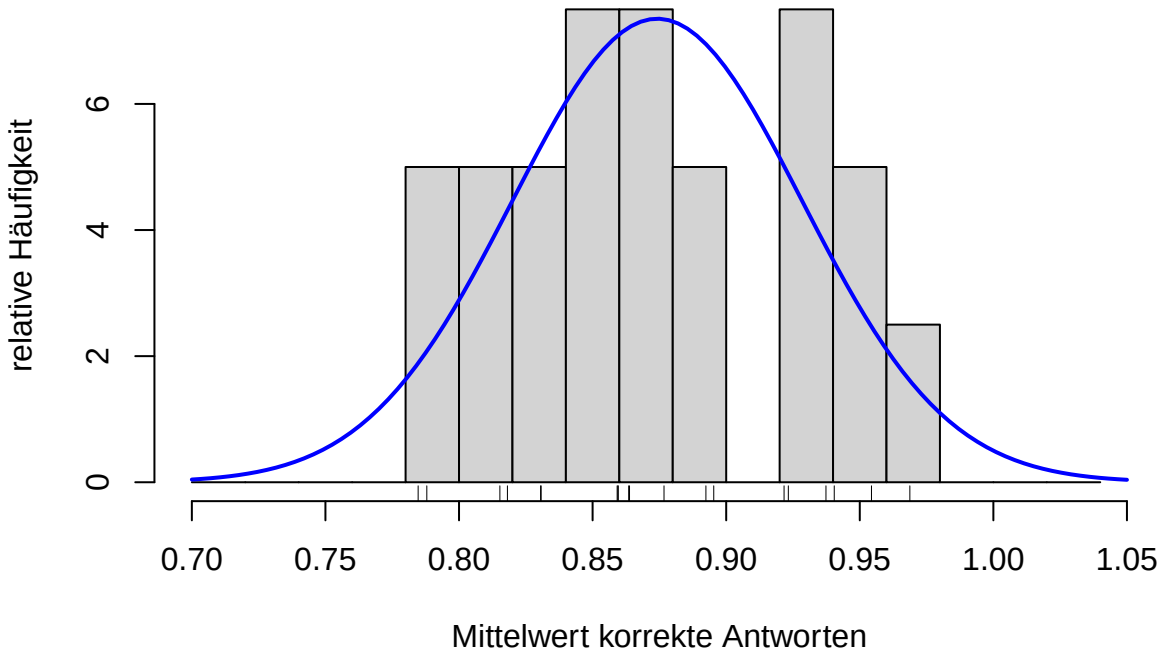

```
colorListResult %>% filter(condition==8,age=="old") -> df_means_o8
fromTo <- round(range(df_means_o8$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_o8$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_o8$mean_answer_correct))
curve(dnorm(x,mean(df_means_o8$mean_answer_correct),sd(df_means_o8$mean_answer_correct)),lwd=2,col="blu
```



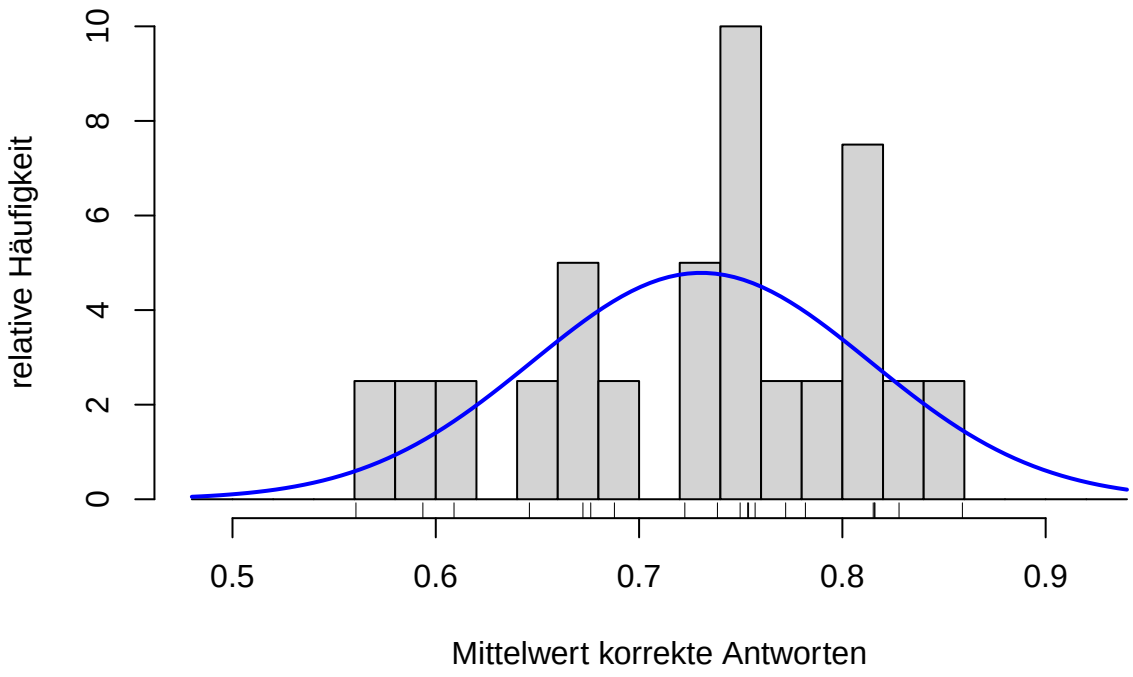

```
colorListResult %>% filter(condition==2,age=="young") -> df_means_y2
fromTo <- round(range(df_means_y2$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_y2$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_y2$mean_answer_correct))
curve(dnorm(x,mean(df_means_y2$mean_answer_correct),sd(df_means_y2$mean_answer_correct)),lwd=2,col="blu
```



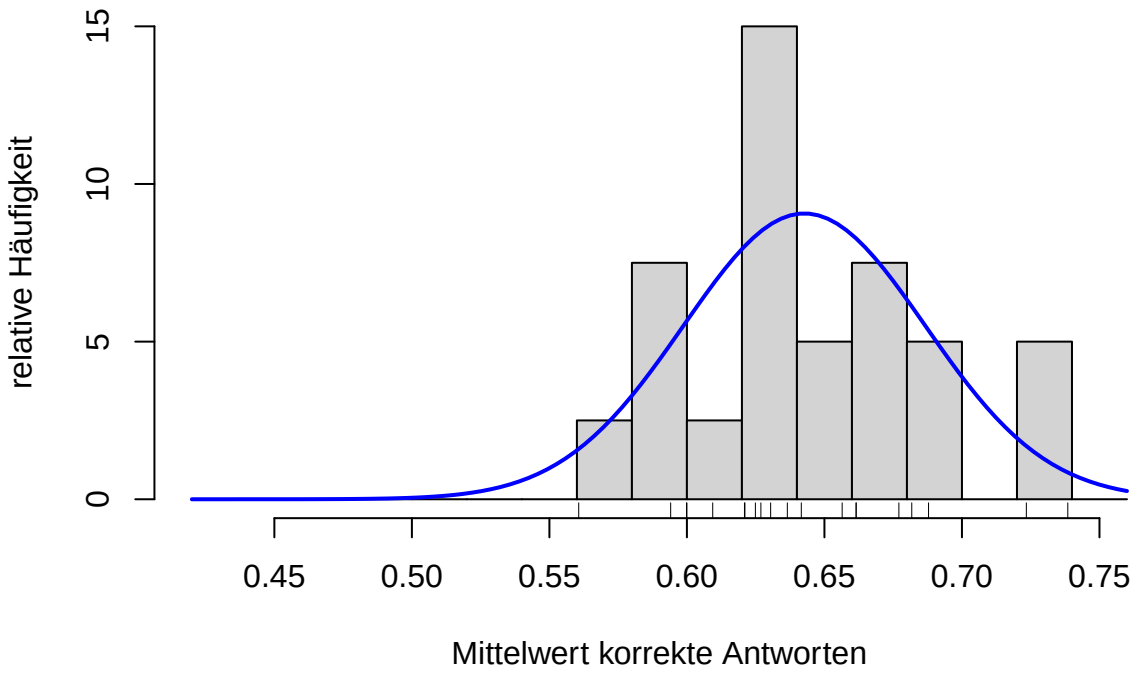

```
colorListResult %>% filter(condition==4,age=="young") -> df_means_y4
fromTo <- round(range(df_means_y4$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_y4$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_y4$mean_answer_correct))
curve(dnorm(x,mean(df_means_y4$mean_answer_correct),sd(df_means_y4$mean_answer_correct)),lwd=2,col="blu
```




```
colorListResult %>% filter(condition==6,age=="young") -> df_means_y6
fromTo <- round(range(df_means_y6$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_y6$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_y6$mean_answer_correct))
curve(dnorm(x,mean(df_means_y6$mean_answer_correct),sd(df_means_y6$mean_answer_correct)),lwd=2,col="blu
```




```
colorListResult %>% filter(condition==8,age=="young") -> df_means_y8
fromTo <- round(range(df_means_o8$mean_answer_correct),2)+c(-0.08,0.08)
limits <- seq(from=fromTo[1], to=fromTo[2], by=0.02)
hist(df_means_y8$mean_answer_correct,freq=FALSE,xlim = fromTo ,xlab = "Mittelwert korrekte Antworten",y
rug(jitter(df_means_y8$mean_answer_correct))
curve(dnorm(x,mean(df_means_y8$mean_answer_correct),sd(df_means_y8$mean_answer_correct)),lwd=2,col="blu
```



K Wert

```
# Color List
colorListResult %>% ungroup() %>% group_by(age,condition) %>%summarise(round(mean(hitRate),2),round(sd(
```

```
## `summarise()` regrouping output by 'age' (override with `.groups` argument)
```

```
names(df_KStat_data) <- c("age","setsize","mean_hit_rate","sd_hit_rate","mean_false_alarm","sd_false_ala
colorListResult %>% ungroup() %>% group_by(age,condition) %>%summarise(paste(round(mean(hitRate),2),"/"
```

```
## `summarise()` regrouping output by 'age' (override with `.groups` argument)
```

```
names(df_KStat) <- c("age","Set Size","HitRate Mean/SD","False Alarm Mean/SD","k Mean (SD)","kPashler M
df_KStat %>% kable(caption = "Überblick Versuchsreihe Color objects K",align = "lcccccr")
```

Table 8: Überblick Versuchsreihe Color objects K

age	Set Size	HitRate Mean/SD	False Alarm Mean/SD	k Mean (SD)	kPashler MEAN/SD
old	2	0.95 / 0.05	0.06 / 0.05	1.78 / 0.15	1.88 / 0.11
old	4	0.82 / 0.11	0.37 / 0.17	1.83 / 0.69	2.85 / 0.78
old	6	0.8 / 0.16	0.47 / 0.2	2.02 / 0.71	4.09 / 1.28
old	8	0.77 / 0.16	0.58 / 0.18	1.51 / 0.82	3.77 / 2.19
young	2	0.99 / 0.02	0.03 / 0.03	1.91 / 0.07	1.97 / 0.04
young	4	0.95 / 0.03	0.21 / 0.11	2.99 / 0.43	3.77 / 0.17
young	6	0.89 / 0.07	0.43 / 0.16	2.76 / 1	4.72 / 0.9
young	8	0.87 / 0.07	0.58 / 0.12	2.27 / 0.72	5.52 / 0.98

```
# Object List
objectResult %>% ungroup() %>% group_by(age,setsize) %>%summarise(round(mean(hitRate),2),round(sd(hitRa
```

```
## `summarise()` regrouping output by 'age' (override with `.groups` argument)
```

```
names(df_O_KStat_data) <- c("age","setsize","mean_hit_rate","sd_hit_rate","mean_false_alarm","sd_false_a
objectResult %>% ungroup() %>% group_by(age,setsize) %>%summarise(paste(round(mean(hitRate),2),"/",round
```

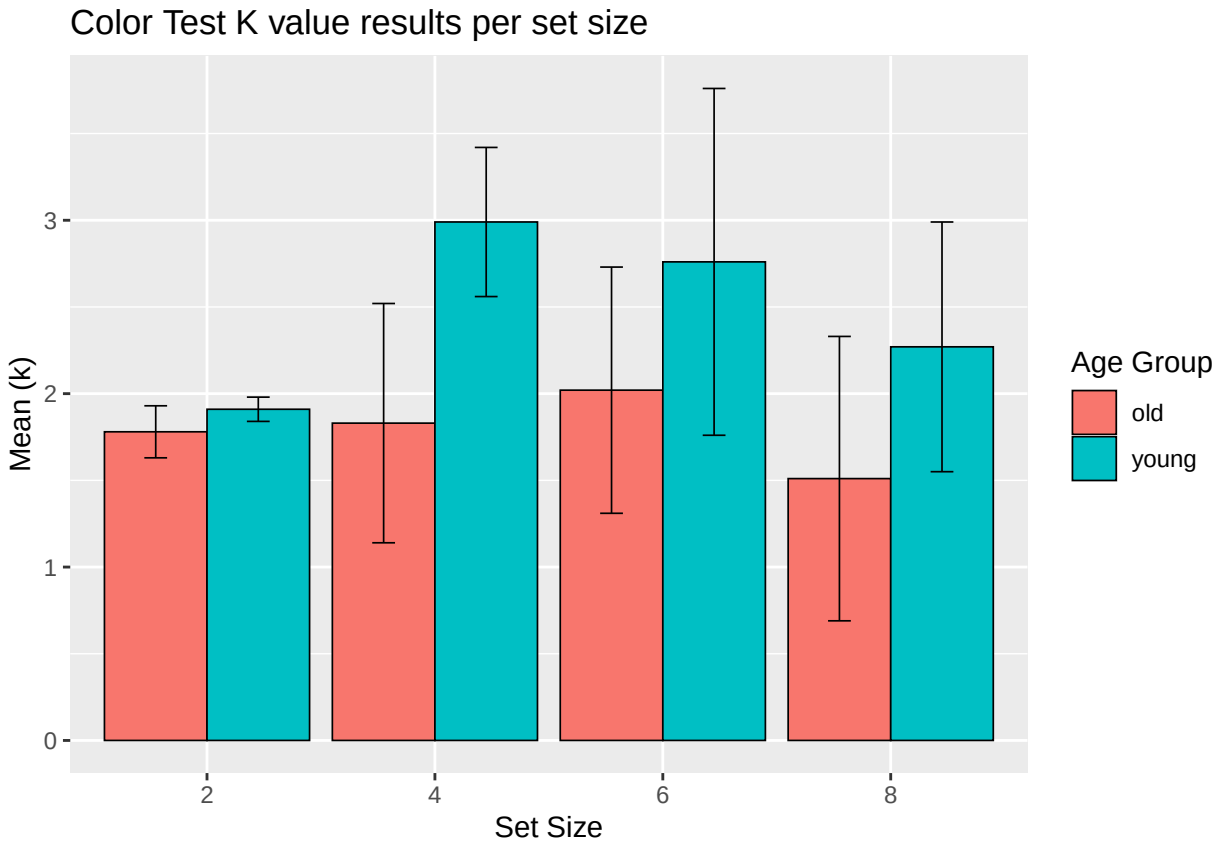
```
## `summarise()` regrouping output by 'age' (override with `.groups` argument)
```

```
names(df_O_KStat) <- c("age","Set Size","HitRate Mean/SD","False Alarm Mean/SD","k Mean (SD)","kPashler
df_O_KStat %>% kable(caption = "Überblick Versuchsreihe Real World Objects K",align = "lcccccr")
```

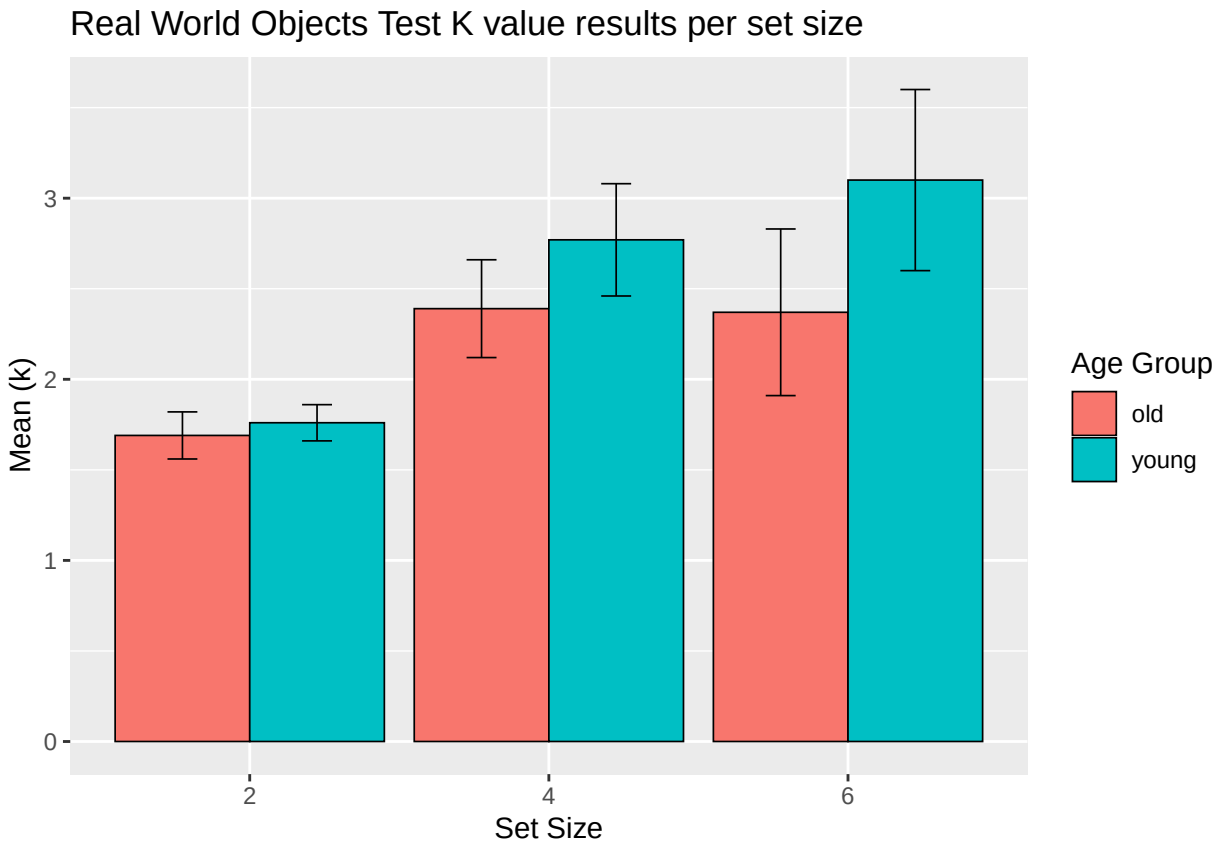
Table 9: Überblick Versuchsreihe Real World Objects K

age	Set Size	HitRate Mean/SD	False Alarm Mean/SD	k Mean (SD)	kPashler MEAN/SD
old	2	0.87 / 0.07	0.03 / 0.03	1.69 / 0.13	1.74 / 0.13
old	4	0.77 / 0.07	0.17 / 0.07	2.39 / 0.27	2.89 / 0.31
old	6	0.75 / 0.12	0.36 / 0.13	2.37 / 0.46	3.76 / 0.85
young	2	0.92 / 0.04	0.04 / 0.03	1.76 / 0.1	1.82 / 0.09
young	4	0.83 / 0.05	0.14 / 0.07	2.77 / 0.31	3.22 / 0.22
young	6	0.82 / 0.07	0.3 / 0.1	3.1 / 0.5	4.47 / 0.52

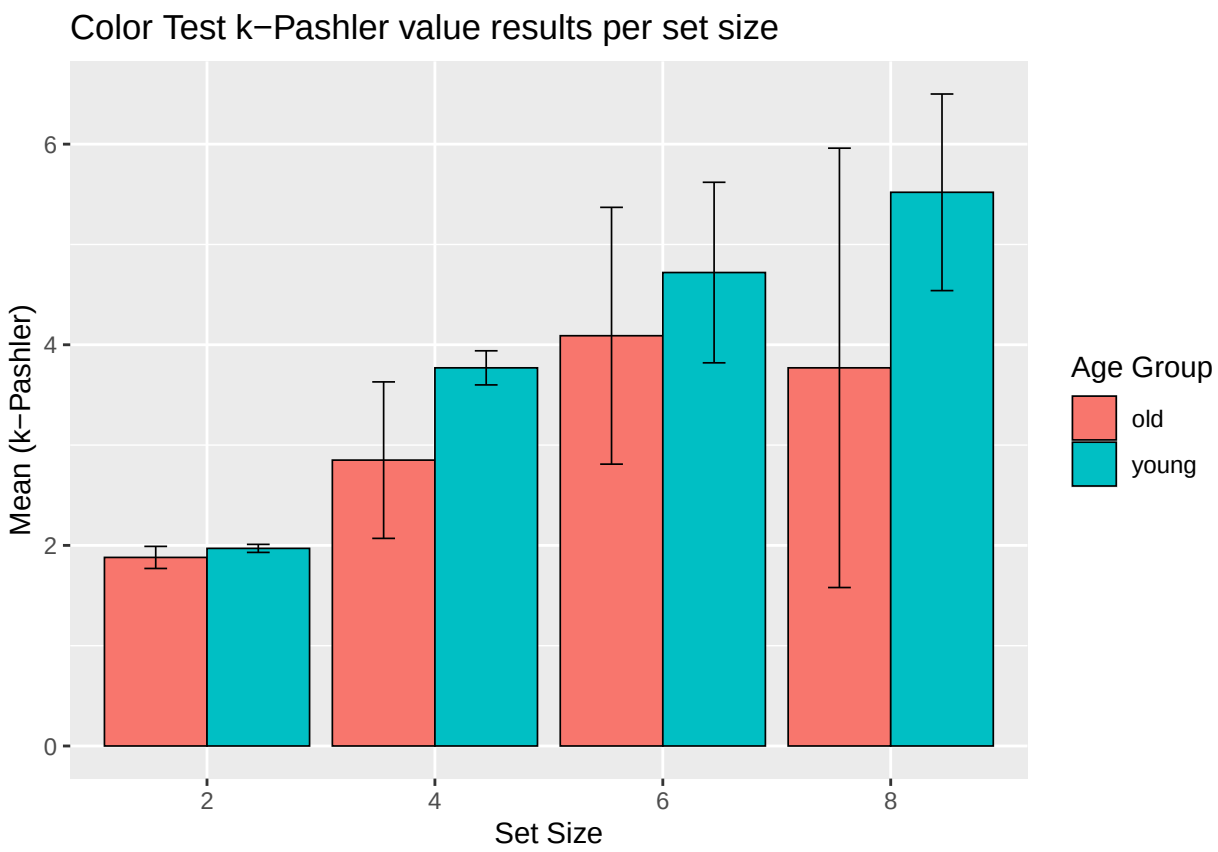
```
#colorListResult %>% ungroup %>% group_by(age,condition) %>% summarise(round(max(k),2)) %>% kable(capti
#colorListResult %>% ungroup %>% group_by(age,condition) %>% summarise(round(max(kPashler),2)) %>% kabl
#objectResult %>% ungroup %>% group_by(age,setsize) %>% summarise(round(max(k),2)) %>% kable(caption =
#objectResult %>% ungroup %>% group_by(age,setsize) %>% summarise(round(max(kPashler),2)) %>% kable(cap

ggplot(df_KStat_data, aes(x=factor(setsize), y=mean_k, fill=factor(age))) + geom_bar(position=position_
```



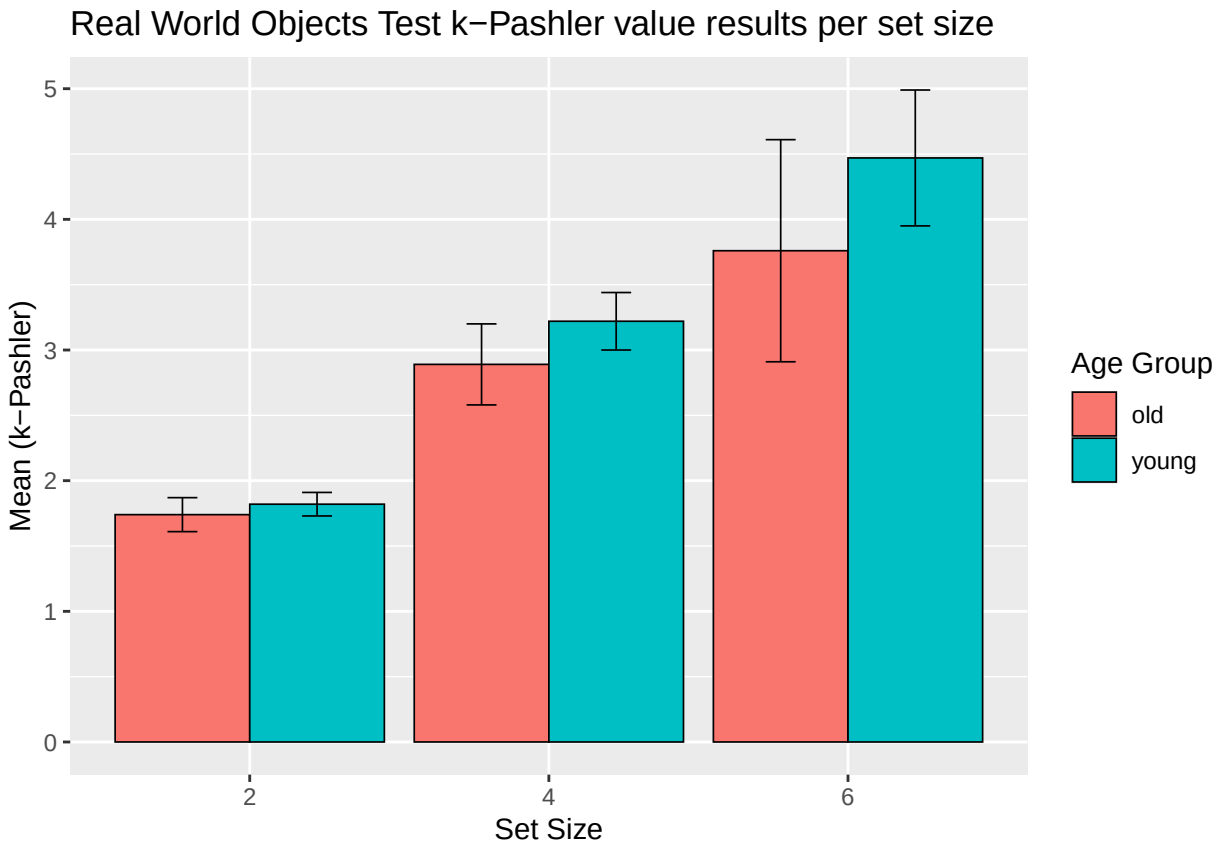
```
ggplot(df_O_KStat_data, aes(x=factor(setsize), y=mean_k, fill=factor(age))) + geom_bar(position=positio
```




```
ggplot(df_KStat_data, aes(x=factor(setsize), y=mean_kPashler, fill=factor(age))) + geom_bar(position=po
```



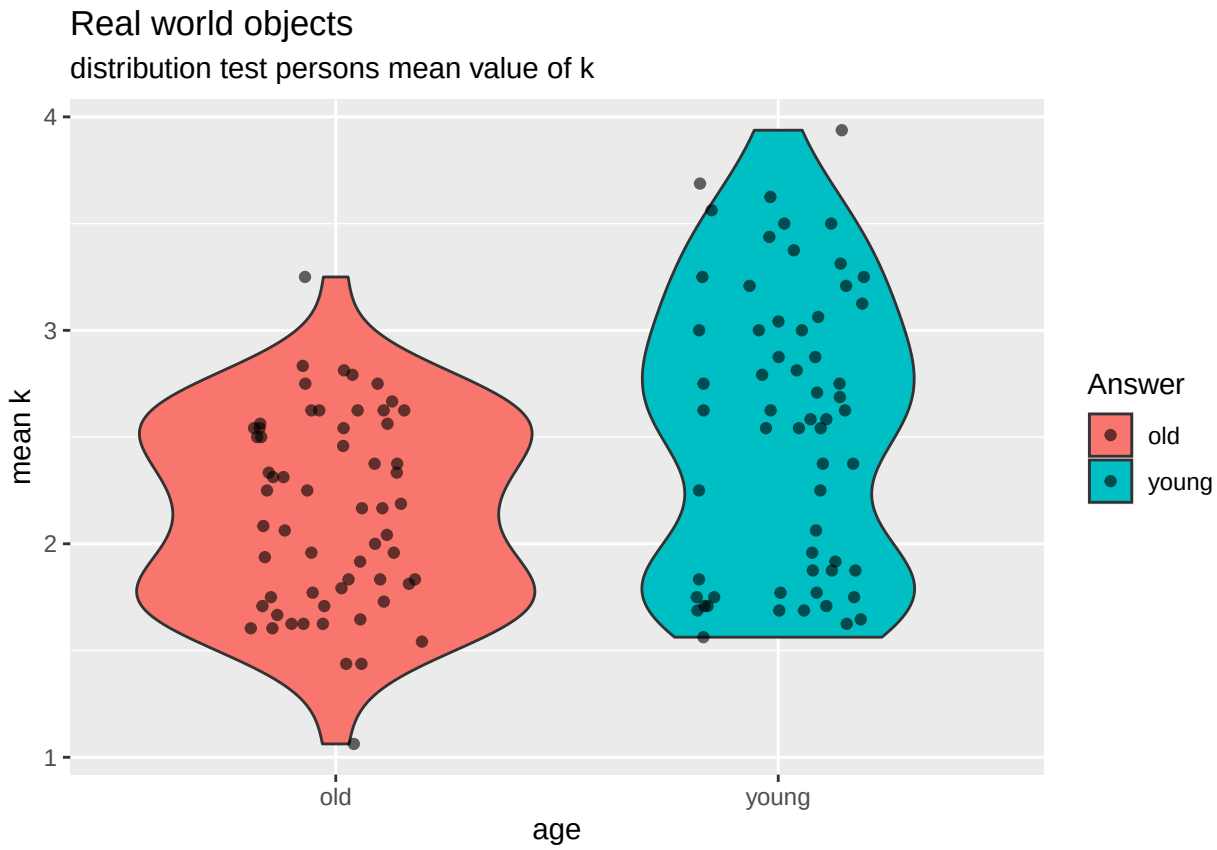

```
ggplot(df_O_KStat_data, aes(x=factor(setsize), y=mean_kPashler, fill=factor(age))) + geom_bar(position=p
```



Grafiken Für K-Werte

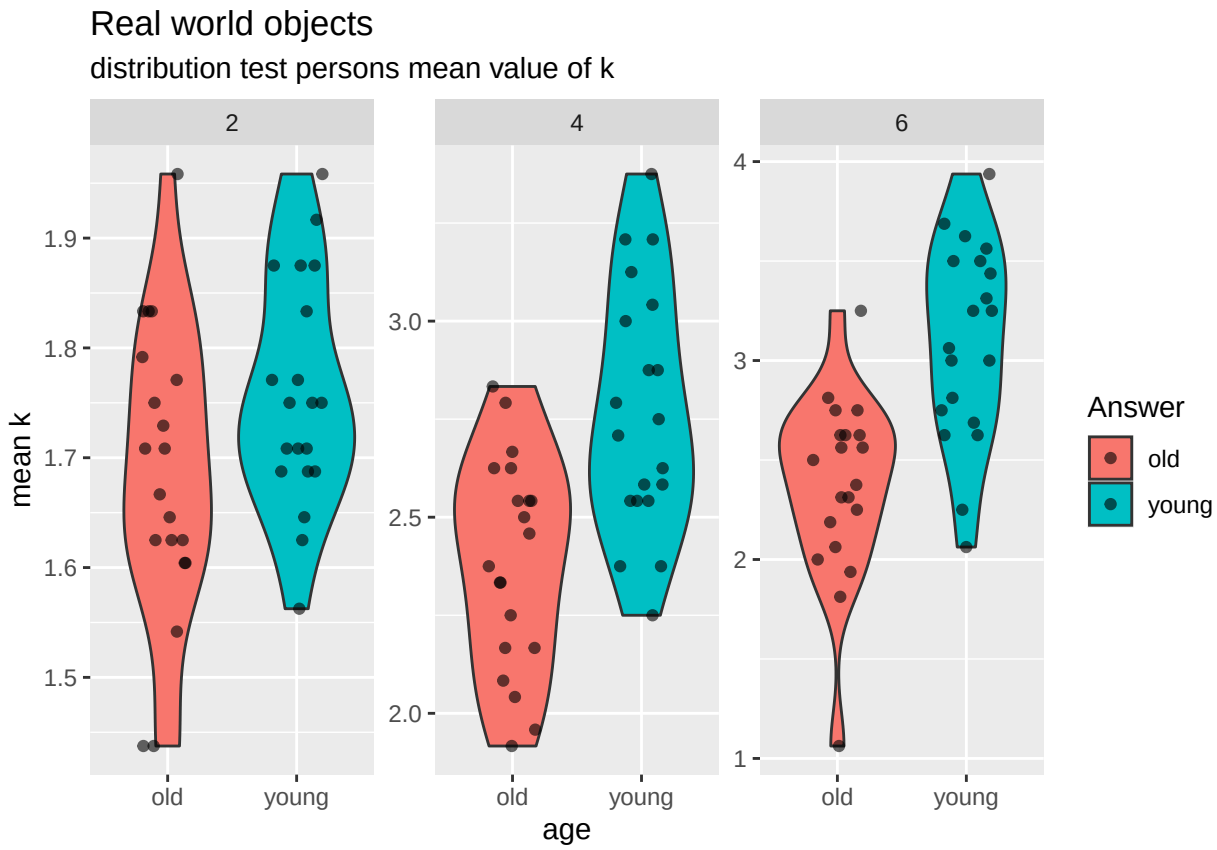
```
colorListResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age)) + geom_violin() + geom_jitter(width = 0.2
```


```
objectResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age)) + geom_violin() + geom_jitter(width = 0.2, al
```



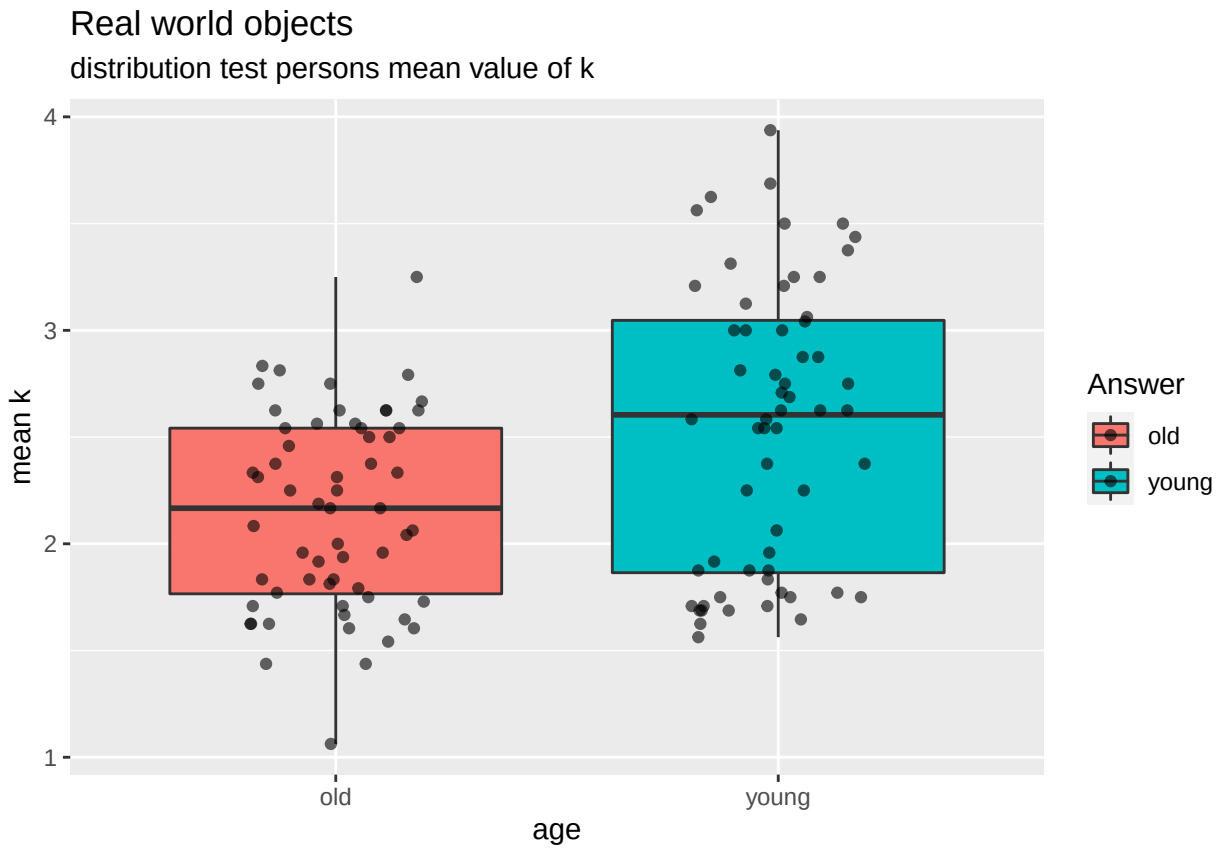

```
colorListResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age))+ geom_violin() + geom_jitter(width = 0.2,
```


```
objectResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age))+ geom_violin() + geom_jitter(width = 0.2, alp
```

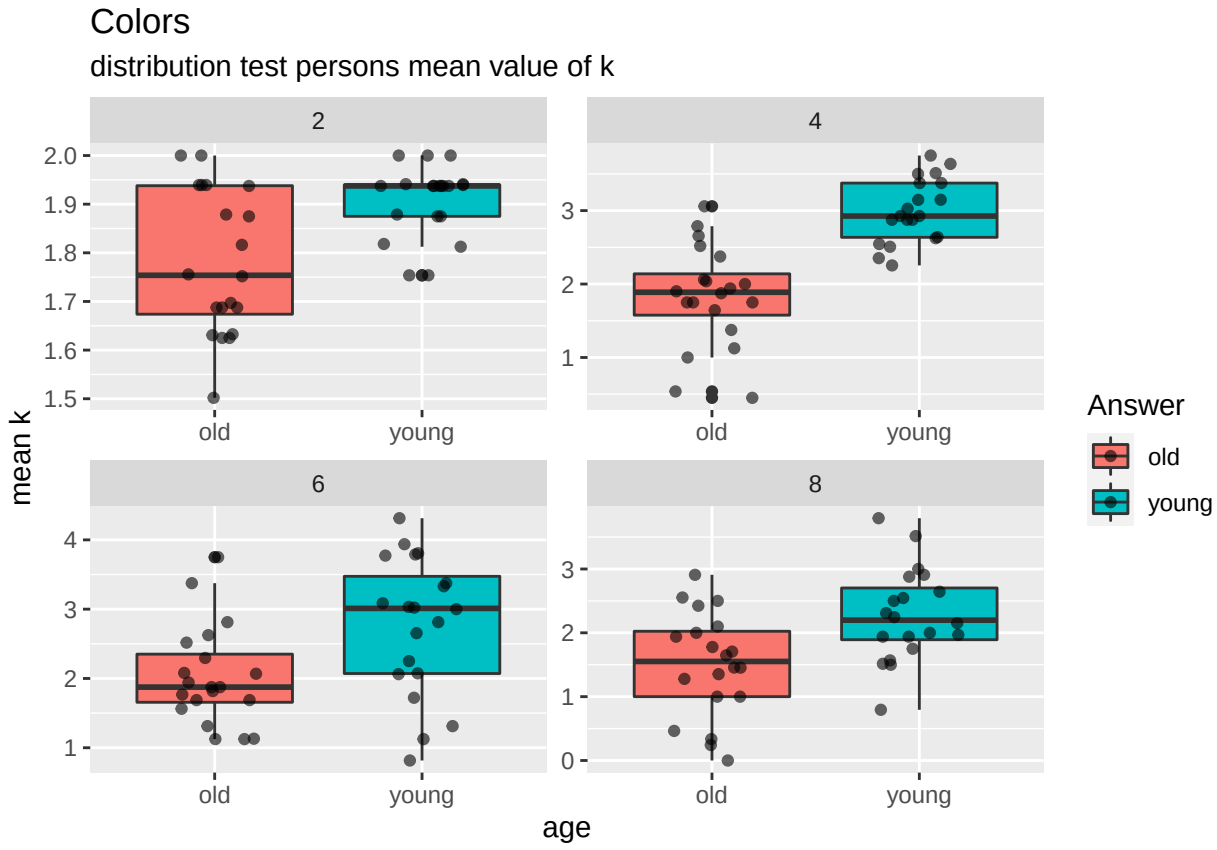



```
colorListResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age)) + geom_boxplot() + geom_jitter(width = 0.1
```

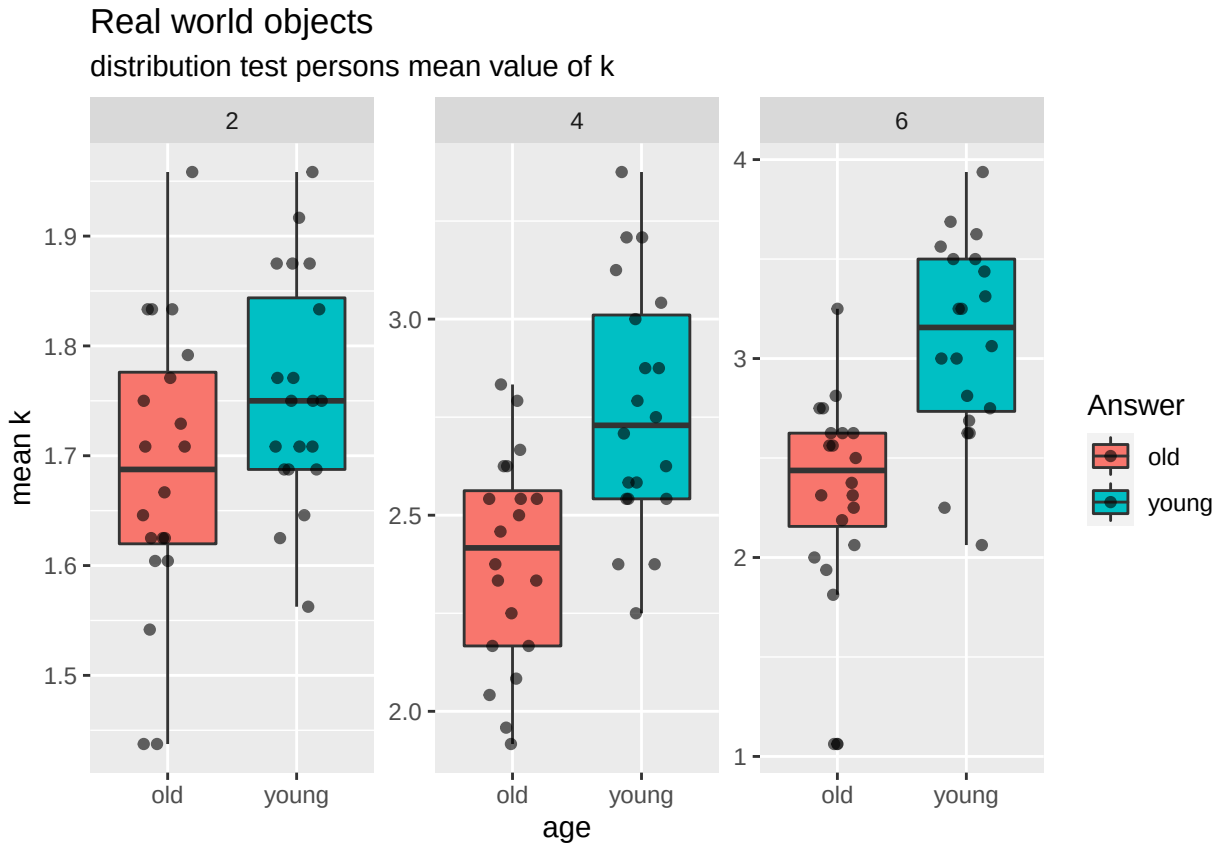

```
objectResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age)) + geom_boxplot() + geom_jitter(width = 0.2, a
```




```
colorListResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age))+ geom_boxplot() + geom_jitter(width = 0.2
```




```
objectResult %>%  ggplot(mapping=aes(x=age, y=k,fill=age))+ geom_boxplot() + geom_jitter(width = 0.2, al
```




```
####### K Barcharts ##########
mkChart <- ggplot(colorListResult, aes(x = condition,y = k, col = factor(age), fill = factor(age)))
```

Unterschied zwischen Colors & Objekte H1: Änderungen von Real World Objekten werden von jungen Personen besser erkannt als Änderungen von Farben. H1: Änderungen von Real World Objekten werden von alten Personen besser erkannt als Änderungen von Farben. Notwendige Auswertungen: T-Test abhängige Stichprobe je alt und jung

Voraussetzungen:

Abhängige Variable ist intervallskaliert - ok Es liegen zwei verbundene Stichproben vor aber die Meßwertpaare sind unabhängig - ok Die Unterschiede zwischen den verbundenen Testwerten sind in der Grundgesamtheit normalverteilt.

```
colorList %>% filter(condition <= 6) %>% summarise(mean(answer_correct)) -> df_mean_color_all
```