Table 1 Examples with Built-in R Datasets

zztable1_nextgen

2025-08-25

Introduction

This vignette demonstrates the versatility of `zztable1_nextgen` using various built-in R datasets. We'll explore different argument combinations and show how the package handles different types of variables and data structures commonly encountered in statistical analysis.

Dataset Examples

1. Motor Trend Car Road Tests (`mtcars`)

The `mtcars` dataset is perfect for demonstrating automotive performance comparisons.

```r
# Prepare mtcars with meaningful factor variables
data(mtcars)
mtcars$transmission <- factor(
  ifelse(mtcars$am == 1, "Manual", "Automatic"),
  levels = c("Automatic", "Manual")
)
mtcars$engine_shape <- factor(
  ifelse(mtcars$vs == 1, "V-shaped", "Straight"),
  levels = c("Straight", "V-shaped")
)
mtcars$cylinders <- factor(mtcars$cyl)

head(mtcars[, c("mpg", "hp", "wt", "transmission", "engine_shape", "cylinders")])
#>                    mpg  hp    wt transmission engine_shape cylinders
#> Mazda RX4         21.0 110 2.620       Manual     Straight         6
#> Mazda RX4 Wag     21.0 110 2.875       Manual     Straight         6
#> Datsun 710        22.8  93 2.320       Manual     V-shaped         4
#> Hornet 4 Drive    21.4 110 3.215    Automatic     V-shaped         6
#> Hornet Sportabout 18.7 175 3.440    Automatic     Straight         8
#> Valiant           18.1 105 3.460    Automatic     V-shaped         6
```

Basic Table by Transmission Type

```r
# Basic table comparing car characteristics by transmission type
bp1 <- table1_nextgen(
  form = transmission ~ mpg + hp + wt + qsec,
  data = mtcars,
  theme = "console"
)
```

```r
# Display the table
display_table(bp1, mtcars)
#> mpg       17.1 (3.8)    24.4 (6.2)    3e-04
#> ==============================================
#>
#> hp        160.3 (53.9)    126.8 (84.1)    0.1798
#>
#> wt        3.8 (0.8)    2.4 (0.6)    0
#>
#> qsec      18.2 (1.8)    17.4 (1.8)    0.2057
#>
```

```r
# Same table with LaTeX formatting for PDF output
bp1_latex <- table1_nextgen(
  form = transmission ~ mpg + hp + wt + qsec,
  data = mtcars,
  layout = "latex",
  theme = "nejm"
)

# Display LaTeX table
display_table(bp1_latex, mtcars)
```

LaTeX Formatted Version

mpg 17.1 (3.8) 24.4 (6.2) 3e-04

hp 160.3 (53.9) 126.8 (84.1) 0.1798

wt 3.8 (0.8) 2.4 (0.6) 0

qsec 18.2 (1.8) 17.4 (1.8) 0.2057

Advanced Table with P-values and Totals

```r
# Advanced table with statistical testing
bp2 <- table1_nextgen(
  form = transmission ~ mpg + hp + wt + cylinders,
  data = mtcars,
  pvalue = TRUE,
  totals = TRUE,
  theme = "nejm"
)

display_table(bp2, mtcars)
#> mpg       17.1 (3.8)    24.4 (6.2)              3e-04
#> ========================================================
#>
#> hp        160.3 (53.9)    126.8 (84.1)              0.1798
#>
#> wt        3.8 (0.8)    2.4 (0.6)              0
#>
#> cylinders
```

```
#> 4          3 (16%)    8 (62%)              0.0091
#> 6          4 (21%)    3 (23%)
#> 8          12 (63%)    2 (15%)
```

```
# Same table with LaTeX formatting for PDF output
bp2_latex <- table1_nextgen(
  form = transmission ~ mpg + hp + wt + cylinders,
  data = mtcars,
  pvalue = TRUE,
  totals = TRUE,
  layout = "latex",
  theme = "nejm"
)

display_table(bp2_latex, mtcars)
```

LaTeX Formatted Version

mpg 17.1 (3.8) 24.4 (6.2) 3e-04

hp 160.3 (53.9) 126.8 (84.1) 0.1798

wt 3.8 (0.8) 2.4 (0.6) 0

cylinders
4 3 (16%) 8 (62%) 0.0091
6 4 (21%) 3 (23%)
8 12 (63%) 2 (15%)

Stratified Analysis by Engine Shape

```
# Stratified analysis
bp3 <- table1_nextgen(
  form = transmission ~ mpg + hp,
  data = mtcars,
  strata = "engine_shape",
  pvalue = TRUE,
  theme = "jama"
)

display_table(bp3, mtcars)
#> mpg        17.1 (3.8)    24.4 (6.2)    3e-04
#> ==============================================
#>
#> hp         160.3 (53.9)    126.8 (84.1)    0.1798
#>
#>
#>
#>
#>
```

```r
# Same stratified analysis with LaTeX formatting
bp3_latex <- table1_nextgen(
  form = transmission ~ mpg + hp,
  data = mtcars,
  strata = "engine_shape",
  pvalue = TRUE,
  layout = "latex",
  theme = "jama"
)

display_table(bp3_latex, mtcars)
```

LaTeX Formatted Version

mpg 17.1 (3.8) 24.4 (6.2) 3e-04

hp 160.3 (53.9) 126.8 (84.1) 0.1798

2. Iris Flower Data (iris)

The classic iris dataset demonstrates biological measurements across species.

```r
data(iris)
# Add a simulated treatment variable for demonstration
set.seed(123)
iris$treatment <- factor(
  sample(c("Control", "Treatment"), nrow(iris), replace = TRUE),
  levels = c("Control", "Treatment")
)

head(iris)
#>   Sepal.Length Sepal.Width Petal.Length Petal.Width Species treatment
#> 1          5.1         3.5          1.4         0.2  setosa   Control
#> 2          4.9         3.0          1.4         0.2  setosa   Control
#> 3          4.7         3.2          1.3         0.2  setosa   Control
#> 4          4.6         3.1          1.5         0.2  setosa Treatment
#> 5          5.0         3.6          1.4         0.2  setosa   Control
#> 6          5.4         3.9          1.7         0.4  setosa Treatment
```

Species Comparison

```r
# Compare measurements across species
bp4 <- table1_nextgen(
  form = Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width,
  data = iris,
  pvalue = TRUE,
  totals = TRUE,
  theme = "lancet"
)

display_table(bp4, iris)
#> Sepal.Length    5.01 (0.35)    5.94 (0.52)    6.59 (0.64)              0
```

```
#> ================================================================================
#>
#> Sepal.Width    3.43 (0.38)    2.77 (0.31)    2.97 (0.32)              0
#>
#> Petal.Length    1.46 (0.17)    4.26 (0.47)    5.55 (0.55)              0
#>
#> Petal.Width    0.25 (0.11)    1.33 (0.2)    2.03 (0.27)              0
#>
```

```
# Same species comparison with LaTeX formatting
bp4_latex <- table1_nextgen(
  form = Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width,
  data = iris,
  pvalue = TRUE,
  totals = TRUE,
  layout = "latex",
  theme = "lancet"
)

display_table(bp4_latex, iris)
```

LaTeX Formatted Version

Sepal.Length 5.01 (0.35) 5.94 (0.52) 6.59 (0.64) 0

Sepal.Width 3.43 (0.38) 2.77 (0.31) 2.97 (0.32) 0

Petal.Length 1.46 (0.17) 4.26 (0.47) 5.55 (0.55) 0

Petal.Width 0.25 (0.11) 1.33 (0.2) 2.03 (0.27) 0

Treatment Groups with Different Numeric Summaries

```
# Custom numeric summary function
median_range <- function(x) {
  paste0(
    format(median(x, na.rm = TRUE), digits = 3),
    " (",
    format(min(x, na.rm = TRUE), digits = 3),
    "-",
    format(max(x, na.rm = TRUE), digits = 3),
    ")"
  )
}

bp5 <- table1_nextgen(
  form = treatment ~ Sepal.Length + Sepal.Width + Species,
  data = iris,
  numeric_summary = median_range,
  pvalue = TRUE,
  theme = "bmj"
)
```

```r
display_table(bp5, iris)
#> Sepal.Length    5.65 (4.3-7.9)    6 (4.4-7.7)    0.0865
#> ==========================================================
#>
#> Sepal.Width    3 (2-4.4)    3 (2.2-4.2)    0.894
#>
#> Species
#> setosa    30 (39%)    20 (27%)    0.0821
#> versicolor    27 (36%)    23 (31%)
#> virginica    19 (25%)    31 (42%)
```

3. Titanic Passenger Data (`Titanic`)

Convert the Titanic table to a data frame for demographic analysis.

```r
# Convert Titanic table to data frame
titanic_df <- as.data.frame(Titanic)
# Expand to individual records
titanic_expanded <- titanic_df[rep(seq_len(nrow(titanic_df)), titanic_df$Freq), -5]

# Add simulated continuous variables for demonstration
set.seed(456)
titanic_expanded$age_est <- ifelse(
  titanic_expanded$Age == "Adult",
  round(rnorm(sum(titanic_expanded$Age == "Adult"), 35, 12)),
  round(rnorm(sum(titanic_expanded$Age == "Child"), 8, 3))
)

titanic_expanded$fare_est <- ifelse(
  titanic_expanded$Class == "1st", rnorm(sum(titanic_expanded$Class == "1st"), 80, 20),
  ifelse(titanic_expanded$Class == "2nd", rnorm(sum(titanic_expanded$Class == "2nd"), 40, 10),
         rnorm(sum(titanic_expanded$Class != "1st" & titanic_expanded$Class != "2nd"), 15, 5))
)

head(titanic_expanded)
#>     Class  Sex   Age Survived age_est  fare_est
#> 3     3rd Male Child       No       7  8.965769
#> 3.1   3rd Male Child       No       9 20.580404
#> 3.2   3rd Male Child       No       6 13.288321
#> 3.3   3rd Male Child       No       6 14.422143
#> 3.4   3rd Male Child       No      11 17.962656
#> 3.5   3rd Male Child       No       8 16.629491
```

Survival Analysis by Class

```r
bp6 <- table1_nextgen(
  form = Survived ~ Class + Sex + Age + age_est + fare_est,
  data = titanic_expanded,
  pvalue = TRUE,
  totals = TRUE,
  missing = TRUE,
  theme = "nejm"
)
```

```r
display_table(bp6, titanic_expanded)
#> Class
#> ================================================
#> 1st       122 (8%)    203 (29%)              [Error]
#> 2nd       167 (11%)    118 (17%)
#> 3rd       528 (35%)    178 (25%)
#> Crew      673 (45%)    212 (30%)
#> Sex
#> Male      1364 (92%)    367 (52%)              0
#> Female    126 (8%)    344 (48%)
#> Age
#> Child     52 (3%)    57 (8%)              0
#> Adult     1438 (97%)    654 (92%)
#> age_est    34.8 (12.6)    33.1 (13.8)              0.0056
#>
#> fare_est    23.1 (20.4)    37.9 (30.5)              0
#>
```

4. Plant Growth Data (`PlantGrowth`)

Experimental data comparing plant weights under different conditions.

```r
data(PlantGrowth)
head(PlantGrowth)
#>   weight group
#> 1   4.17  ctrl
#> 2   5.58  ctrl
#> 3   5.18  ctrl
#> 4   6.11  ctrl
#> 5   4.50  ctrl
#> 6   4.61  ctrl

# Simple treatment comparison
bp7 <- table1_nextgen(
  form = group ~ weight,
  data = PlantGrowth,
  pvalue = TRUE,
  totals = TRUE,
  numeric_summary = "median_iqr",  # Built-in alternative summary
  theme = "console"
)

display_table(bp7, PlantGrowth)
#> weight    5.2 [4.6-5.3]    4.6 [4.2-4.9]    5.4 [5.3-5.7]              0.1944
#> ==============================================================================
#>
```

5. Tooth Growth Data (`ToothGrowth`)

Guinea pig tooth growth under different vitamin C treatments.

```r
data(ToothGrowth)
ToothGrowth$dose <- factor(ToothGrowth$dose)
head(ToothGrowth)
```

```
#>    len supp dose
#> 1  4.2   VC  0.5
#> 2 11.5   VC  0.5
#> 3  7.3   VC  0.5
#> 4  5.8   VC  0.5
#> 5  6.4   VC  0.5
#> 6 10.0   VC  0.5

# Compare by supplement type
bp8 <- table1_nextgen(
  form = supp ~ len + dose,
  data = ToothGrowth,
  pvalue = TRUE,
  totals = TRUE,
  theme = "jama"
)

display_table(bp8, ToothGrowth)
#> len        20.7 (6.6)    17 (8.3)              0.0604
#> ======================================================
#>
#> dose
#> 0.5        10 (33%)    10 (33%)              1
#> 1          10 (33%)    10 (33%)
#> 2          10 (33%)    10 (33%)
```

Stratified by Dose

```
bp9 <- table1_nextgen(
  form = supp ~ len,
  data = ToothGrowth,
  strata = "dose",
  pvalue = TRUE,
  theme = "lancet"
)

display_table(bp9, ToothGrowth)
#> len        20.66 (6.61)    16.96 (8.27)    0.0604
#> ==================================================
#>
#>
#>
#>
#>
```

6. Chickwts Data (Chicken Weights)

Chicken weights by different feed types.

```
data(chickwts)
head(chickwts)
#>   weight      feed
#> 1    179 horsebean
#> 2    160 horsebean
```

```
#> 3    136 horsebean
#> 4    227 horsebean
#> 5    217 horsebean
#> 6    168 horsebean

bp10 <- table1_nextgen(
  form = feed ~ weight,
  data = chickwts,
  pvalue = TRUE,
  totals = TRUE,
  theme = "bmj"
)

display_table(bp10, chickwts)
#> weight    323.6 (64.4)    160.2 (38.6)    218.8 (52.2)    276.9 (64.9)    246.4 (54.1)    328.9 (48.
#> ==========================================================================================================
#>
```

7. Built-in Dataset with Missing Values (`airquality`)

Environmental data with naturally occurring missing values.

```
data(airquality)
airquality$Month <- factor(
  month.name[airquality$Month],
  levels = month.name[5:9]  # May through September
)
head(airquality)
#>   Ozone Solar.R Wind Temp Month Day
#> 1    41     190  7.4   67   May   1
#> 2    36     118  8.0   72   May   2
#> 3    12     149 12.6   74   May   3
#> 4    18     313 11.5   62   May   4
#> 5    NA      NA 14.3   56   May   5
#> 6    28      NA 14.9   66   May   6

# Show how missing values are handled
bp11 <- table1_nextgen(
  form = Month ~ Ozone + Solar.R + Wind + Temp,
  data = airquality,
  missing = TRUE,  # Include missing value rows
  pvalue = TRUE,
  totals = TRUE,
  theme = "nejm"
)

display_table(bp11, airquality)
#> Ozone     23.6 (22.2)    29.4 (18.2)    59.1 (31.6)    60 (39.7)    31.4 (24.1)              0.6088
#> ==========================================================================================================
#>
#>
#> Solar.R    181.3 (115.1)    190.2 (92.9)    216.5 (80.6)    171.9 (76.8)    167.4 (79.1)
#>
#>
```

```
#> Wind      11.6 (3.5)    10.3 (3.8)    8.9 (3)    8.8 (3.2)    10.2 (3.5)              0.1228
#>
#> Temp      65.5 (6.9)    79.1 (6.6)    83.9 (4.3)    84 (6.6)    76.9 (8.4)              0
#>
```

Argument Combinations Summary

Key Arguments Demonstrated

1. **`pvalue`**: Statistical testing between groups
2. **`totals`**: Include total column
3. **`missing`**: Handle and display missing values
4. **`strata`**: Stratified analysis
5. **`numeric_summary`**: Different summary statistics
6. **`theme`**: Various journal and console themes

Supported Numeric Summaries

- **Built-in options**:
  - `"mean_sd"` (default): Mean ± Standard Deviation
  - `"median_iqr"`: Median [Q1, Q3]
- **Custom functions**: Any function that takes a numeric vector and returns a character string

Themes Available

- `"console"`: Plain text output
- `"nejm"`: New England Journal of Medicine style
- `"lancet"`: The Lancet style
- `"jama"`: JAMA style

- `"bmj"`: BMJ style

Performance Notes

The `zztable1_nextgen` package uses several optimizations:

- **Sparse storage**: Only populated cells are stored in memory
- **Vectorized operations**: Efficient processing of multiple variables
- **Lazy evaluation**: Computations performed only when needed
- **Modular architecture**: Clean separation of concerns

```
# Demonstrate with larger simulated dataset
set.seed(789)
large_data <- data.frame(
  treatment = factor(sample(c("Placebo", "Drug A", "Drug B"), 1000, replace = TRUE)),
  age = round(rnorm(1000, 65, 15)),
  sex = factor(sample(c("Male", "Female"), 1000, replace = TRUE)),
  weight = round(rnorm(1000, 70, 15), 1),
  height = round(rnorm(1000, 170, 10), 1),
  center = factor(sample(paste("Center", 1:5), 1000, replace = TRUE))
)

# Time the table creation
system.time({
```

```r
  bp_large <- table1_nextgen(
    form = treatment ~ age + sex + weight + height,
    data = large_data,
    strata = "center",
    pvalue = TRUE,
    totals = TRUE,
    theme = "nejm"
  )
})
#>    user  system elapsed
#>   0.002   0.000   0.003

# Check memory usage
format(object.size(bp_large), units = "KB")
#> [1] "142.1 Kb"
```

Conclusion

The zztable1_nextgen package provides a flexible and efficient way to create publication-ready "Table 1" summaries. The examples in this vignette demonstrate:

- Versatility across different data types and structures
- Comprehensive argument combinations
- Performance with larger datasets
- Clean, readable output suitable for various publication formats

The package maintains the familiar R formula interface while providing significant performance improvements and enhanced functionality through its optimized architecture.