

Package: nawtilus (via r-universe)

September 9, 2024

Type Package

Title Navigated Weighting for the Inverse Probability Weighting

Version 0.1.4

Description Implements the navigated weighting (NAWT) proposed by Katsumata (2020) <[arXiv:2005.10998](https://arxiv.org/abs/2005.10998)>, which improves the inverse probability weighting by utilizing estimating equations suitable for a specific pre-specified parameter of interest (e.g., the average treatment effects or the average treatment effects on the treated) in propensity score estimation. It includes the covariate balancing propensity score proposed by Imai and Ratkovic (2014) <[doi:10.1111/rssb.12027](https://doi.org/10.1111/rssb.12027)>, which uses covariate balancing conditions in propensity score estimation. The point estimate of the parameter of interest as well as coefficients for propensity score estimation and their uncertainty are produced using the M-estimation. The same functions can be used to estimate average outcomes in missing outcome cases.

License GPL-3

Encoding UTF-8

LazyData true

Imports MASS,

RoxygenNote 7.1.0

Depends R (>= 2.10)

Suggests hypergeo, testthat

Repository <https://hirotokatsumata.r-universe.dev>

RemoteUrl <https://github.com/hirotokatsumata/nawtilus>

RemoteRef HEAD

RemoteSha 1af124fe11d82bd81759648f37488216a5270fb2

Contents

cbcheck	2
LaLonde	4
nawt	5
plot.nawt	10
plot_omega	11
print.nawt	13
summary.nawt	14

Index	15
--------------	-----------

cbcheck	<i>Summarize and plot covariate balance</i>
---------	---------------------------------------------

Description

Summarizes and plots covariate balance between treatment and control groups before and after the navigated weighting.

Usage

```
cbcheck(
  object,
  addcov = NULL,
  standardize = TRUE,
  plot = TRUE,
  absolute = TRUE,
  threshold = 0,
  sort = TRUE
)
```

Arguments

object	an object of class “nawt”, usually, a result of a call to nawt .
addcov	a one-sided formula specifying additional covariates whose balance is checked. Covariates containing NAs are automatically dropped.
standardize	a logical value indicating whether weighted mean differences are standardized or not.
plot	a logical value indicating whether a covariate balance plot is displayed.
absolute	a logical value indicating whether the absolute values of differences in weighted means are used in the covariate balance plot.
threshold	an optional numeric vector used as threshold markers in the covariate balance plot.
sort	a logical value indicating whether covariates in the covariate balance plot are sorted by the values of differences in the weighted means before the navigated weighting.

Details

Position of the legend is determined internally.

Value

A matrix whose rows are the covariates and columns are the differences in the (un)standardized weighted mean between the treatment and control groups before (`diff.un`) and after (`diff.adj`) the navigated weighting. The standardized weighted mean is the weighted mean divided by the standard deviation of the covariate for the target population (the treatment group for the average treatment effects on the treated estimation and the whole population for the other quantity of interest). The differences in the categorical variables are not standardized.

Author(s)

Hiroto Katsumata

Examples

```
# Simulation from Kang and Shafer (2007) and Imai and Ratkovic (2014)
# ATT estimation
# True ATT is 10
tau <- 10
set.seed(12345)
n <- 1000
X <- matrix(rnorm(n * 4, mean = 0, sd = 1), nrow = n, ncol = 4)
prop <- 1 / (1 + exp(X[, 1] - 0.5 * X[, 2] + 0.25 * X[, 3] + 0.1 * X[, 4]))
treat <- rbinom(n, 1, 1 - prop)
y <- 210 + 27.4 * X[, 1] + 13.7 * X[, 2] + 13.7 * X[, 3] + 13.7 * X[, 4] +
  tau * treat + rnorm(n)
df <- data.frame(X, treat, y)
colnames(df) <- c("x1", "x2", "x3", "x4", "treat", "y")

# A misspecified model
Xmis <- data.frame(x1mis = exp(X[, 1] / 2),
  x2mis = X[, 2] * (1 + exp(X[, 1]))^(-1) + 10,
  x3mis = (X[, 1] * X[, 3] / 25 + 0.6)^3,
  x4mis = (X[, 2] + X[, 4] + 20)^2)

# Data frame and a misspecified formula for propensity score estimation
df <- data.frame(df, Xmis)
formula_m <- as.formula(treat ~ x1mis + x2mis + x3mis + x4mis)

# Misspecified propensity score model
# Power weighting function with alpha = 2
fits2m <- nawt(formula = formula_m, outcome = "y", estimand = "ATT",
  method = "score", data = df, alpha = 2)
cbcheck(fits2m, addcov = ~ x1 + x2 + x3 + x4)

# Covariate balancing weighting function
fitcbm <- nawt(formula = formula_m, outcome = "y", estimand = "ATT",
  method = "cb", data = df)
```

```

cbcheck(fitcbm, addcov = ~ x1 + x2 + x3 + x4)

# Standard logistic regression
fits0m <- nawt(formula = formula_m, outcome = "y", estimand = "ATT",
              method = "score", data = df, alpha = 0)
cbcheck(fits0m, addcov = ~ x1 + x2 + x3 + x4)

# Display the covariate balance matrix
cb <- cbcheck(fits2m, addcov = ~ x1 + x2 + x3 + x4, plot = FALSE)
cb

```

LaLonde

LaLonde data set

Description

Data from the National Supported Work Demonstration Program and the Panel Study for Income Dynamics. A benchmark data set, which is extensively analyzed by LaLonde (1986), Dehejia and Wahba (1999), and Imai and Ratkovic (2014).

Usage

LaLonde

Format

A data frame with 3212 observations and 12 variables.

exper an indicator for whether the observed unit was in the experimental subset

treat an indicator for whether the individual received the treatment

age age in years

educ schooling in years

black an indicator for black

hisp an indicator for Hispanic

married an indicator for marriage status, one for married

nodegr an indicator for no high school degree

re74 reported earnings in 1974

re75 reported earnings in 1975

re78 reported earnings in 1978

re74.miss an indicator for whether the 1974 earnings variable is missing

Data not missing 1974 earnings are the Dehejia-Wahba subsample of the LaLonde data. Missing values for 1974 earnings set to zero. 1974 and 1975 earnings are pre-treatment. 1978 earnings is taken as the outcome variable.

Source

This version of the data is from [LaLonde](#) data in CBPS package.

References

Dehejia, Rajeev H., and Sadek Wahba. 1999. "Causal Effects in Nonexperimental Studies: Reevaluating the Evaluation of Training Programs." *Journal of the American Statistical Association* 94 (448): 1053–62.

Imai, Kosuke and Marc Ratkovic. 2014. "Covariate Balancing Propensity Score." *Journal of the Royal Statistical Society, Series B (Statistical Methodology)* 76 (1): 243–63.

LaLonde, Robert J. 1986. "Evaluating the Econometric Evaluations of Training Programs with Experimental Data." *American Economic Review* 76 (4): 604–20.

nawt

Navigated weighting (NAWT) estimation

Description

nawt estimates a pre-specified parameter of interest (e.g., the average treatment effects (ATE) or the average treatment effects on the treated (ATT)) with the inverse probability weighting where propensity scores are estimated using estimating equations suitable for the parameter of interest. It includes the covariate balancing propensity score proposed by Imai and Ratkovic (2014), which uses covariate balancing conditions in propensity score estimation. nawt can also be used to estimate average outcomes in missing outcome cases.

Usage

```
nawt(  
  formula,  
  outcome,  
  estimand = "ATT",  
  method = "score",  
  data,  
  weights = NULL,  
  alpha = 2,  
  twostep = TRUE,  
  boot = FALSE,  
  B = 2000,  
  clevel = 0.95,  
  message = TRUE  
)
```

Arguments

formula	an object of class <code>formula</code> (or one that can be coerced to that class): a symbolic description of the model to be fitted.
outcome	a character string specifying the name of outcome values in data.
estimand	a character string specifying a parameter of interest. Choose "ATT" for the average treatment effects on the treated estimation, "ATE" for the average treatment effects estimation, "ATC" for the average outcomes estimation in missing outcome cases. You can choose "ATEcombined" for the combined estimation for the average treatment effects estimation.
method	a character string specifying a type of weighting functions in propensity score estimation ($\omega(\pi)$). Choose "score" for a power function of propensity scores (need to specify the value for alpha), "cb" for a covariate balancing weighting function, or "both" to use both the above weighting functions (need to specify the value for alpha).
data	a data frame (or one that can be coerced to that class) containing the outcomes and the variables in the model.
weights	an optional vector of 'prior weights' (e.g. sampling weights) to be used in the fitting process. Should be NULL or a numeric vector.
alpha	a positive value for an exponent in a power weighting function ($\omega(\pi) = \pi^\alpha$, in the ATT estimation, for example). Default is 2. Set to 0 to use the standard logistic regression for propensity score estimation. Note that nawt with alpha being one of the pre-specified values (0, 0.5, 1, ..., 5) runs substantially faster than with any other values, and the latter case requires hypergeo package.
twostep	a logical value indicating whether to use a two-step estimator when method = "both". Default is TRUE. Set to FALSE to use a continuously-updating GMM estimator, which is substantially computationally intensive.
boot	a logical value indicating whether to use a non-parametric bootstrapping method to estimate the variance-covariance matrix and confidence intervals for parameters. Default is FALSE. Set to FALSE to use a sandwich-type asymptotic covariance estimator.
B	the number of bootstrap replicates. Default is 2,000.
clevel	confidence level. Default is 0.95.
message	a logical value indicating whether messages are shown or not.

Details

The treatment variable (or, missingness variable in missing outcome cases) must be binary and coded as 0 (for controlled or non-missing observations) or 1 (for treated or missing observations).

When the data frame has incomplete cases, which have NAs for either of the treatment variable, explanatory variables for propensity score estimation, or the outcome variable, nawt conducts listwise deletion. Returned values (e.g., weights, ps, data) do not contain values for these deleted cases.

The parameter of interest is estimated by the Hajek estimator, where inverse probability weights are standardized to sum to 1 within each treatment group after being calculated as $t_i/\pi_i - (1 - t_i)/(1 - \pi_i)$ for the ATE estimation, $(t_i - \pi_i)/(1 - \pi_i)$ for the ATT estimation, $(t_i - \pi_i)/\pi_i$ for the ATC estimation, and $(1 - t_i)/(1 - \pi_i)$ for the missing outcome cases.

For the ATE estimation, it is recommended to specify the estimand as "ATE" but you may specify it as "ATEcombined". The former utilizes the separated estimation whereas the latter utilizes the combined estimation, and the former should produce smaller biases and variances. Note that the former estimates two propensity scores for each observation by estimating two propensity score functions with different estimating equations.

When a two-step estimator is used in `nawt` with `method = "both"`, `scratio` (r) is calculated in the first step. `scratio` is a ratio of accuracy in propensity score estimation in the NAWT with a power weighting function with a specified `alpha` to that with a covariate balancing weighting function. It determines the mixture weight in the second step, like the weighting matrix in the two-step over-identified GMM estimation, where weighted estimating equations of those with the power weighting function and the covariate balancing function is used. This mixture weight is proportional to the `scratio` (e.g., $\omega(\pi) = r\pi^\alpha + (1 - r)/(1 - \pi)$, in the ATT estimation).

Since the NAWT utilizes weighted estimating equations in propensity score estimation, it may sometimes become unstable especially when only a few observations have extremely large weights in propensity score estimation. `nawt` generates a warning when the effective sample size for propensity score estimation is smaller than a quarter of the effective sample size with the initial weights. In that case, carefully look at the estimated coefficients to check whether the estimation fails or not and `cbcheck` will be helpful.

Value

`nawt` returns an object of class inheriting from "nawt".

The function summary (i.e., `summary.nawt`) can be used to obtain or print a summary of the results.

An object of class "nawt" is a list containing the following components:

<code>est</code>	the point estimate of the parameter of interest.
<code>weights</code>	the estimated inverse probability weights.
<code>ps</code>	the estimated propensity scores. A matrix of two sets of the estimated propensity scores is returned when <code>estimand = "ATE"</code> .
<code>coefficients</code>	a named vector of coefficients. A matrix of two sets of coefficients for two sets of propensity scores is returned when <code>estimand = "ATE"</code> .
<code>varcov</code>	the variance-covariance matrix of the coefficients and parameter of interest.
<code>converged</code>	logical. Was the algorithm judged to have converged?
<code>naive_weights</code>	the estimated inverse probability weights with the standard logistic regression for the propensity score estimation.
<code>naive_coef</code>	a named vector of coefficients with the standard logistic regression for the propensity score estimation.
<code>scratio</code>	an optimal ratio of the covariate balancing weighting function to the power weighting function in taking the weighted average weights for the weighted score conditions when <code>method = "both"</code> and <code>twostep = TRUE</code> . A vector of length two for two propensity score estimation is returned when <code>estimand = "ATE"</code> .
<code>estimand</code>	the parameter of interest specified.
<code>method</code>	the method specified.
<code>outcome</code>	the outcome vector.

alpha	alpha specified.
names.x	names of the explanatory variables in propensity score estimation.
prior.weights	the weights initially supplied, a vector of 1s if none were.
treat	the treatment vector. The missingness vector when the missing outcome cases.
ci	a matrix of the confidence intervals for the parameter of interest.
omega	a vector of weights for the weighted score conditions (ω). A matrix of two sets of omega is returned when estimand = "ATE".
effN_ps	the effective sample size for the propensity score estimation. A vector of length two for two propensity score estimation is returned when estimand = "ATE".
effN_est	the effective sample size for the parameter of interest estimation.
effN_original	the effective sample size with the initial weights.
formula	formula specified.
call	the matched call.
data	the data argument.

Author(s)

Hiroto Katsumata

References

- Imai, Kosuke and Marc Ratkovic. 2014. "Covariate Balancing Propensity Score." *Journal of the Royal Statistical Society, Series B (Statistical Methodology)* 76 (1): 243–63.
- Christian Fong, Marc Ratkovic and Kosuke Imai (2019). CBPS: Covariate Balancing Propensity Score. R package version 0.21. <https://CRAN.R-project.org/package=CBPS>
- Katsumata, Hiroto. 2020. "Navigated Weighting to Improve Inverse Probability Weighting for Missing Data Problems and Causal Inference." arXiv preprint arXiv:2005.10998.

See Also

[summary.nawt](#)

Examples

```
# Simulation from Kang and Shafer (2007) and Imai and Ratkovic (2014)
# ATT estimation
# True ATT is 10
tau <- 10
set.seed(12345)
n <- 1000
X <- matrix(rnorm(n * 4, mean = 0, sd = 1), nrow = n, ncol = 4)
prop <- 1 / (1 + exp(X[, 1] - 0.5 * X[, 2] + 0.25 * X[, 3] + 0.1 * X[, 4]))
treat <- rbinom(n, 1, prop)
y <- 210 + 27.4 * X[, 1] + 13.7 * X[, 2] + 13.7 * X[, 3] + 13.7 * X[, 4] +
  tau * treat + rnorm(n)
df <- data.frame(X, treat, y)
```



```

colnames(df) <- c("x1", "x2", "x3", "x4", "treat", "y")

# A misspecified model
Xmis <- data.frame(x1mis = exp(X[, 1] / 2),
                  x2mis = X[, 2] * (1 + exp(X[, 1]))^(-1) + 10,
                  x3mis = (X[, 1] * X[, 3] / 25 + 0.6)^3,
                  x4mis = (X[, 2] + X[, 4] + 20)^2)

# Data frame and formulas for propensity score estimation
df <- data.frame(df, Xmis)
formula_c <- as.formula(treat ~ x1 + x2 + x3 + x4)
formula_m <- as.formula(treat ~ x1mis + x2mis + x3mis + x4mis)

# Correct propensity score model
# Power weighting function with alpha = 2
fits2c <- nawt(formula = formula_c, outcome = "y", estimand = "ATT",
               method = "score", data = df, alpha = 2)
summary(fits2c)

# Covariate balancing weighting function
fitcbc <- nawt(formula = formula_c, outcome = "y", estimand = "ATT",
               method = "cb", data = df)
summary(fitcbc)

# Standard logistic regression
fits0c <- nawt(formula = formula_c, outcome = "y", estimand = "ATT",
               method = "score", data = df, alpha = 0)
summary(fits0c)

# Misspecified propensity score model
# Power weighting function with alpha = 2
fits2m <- nawt(formula = formula_m, outcome = "y", estimand = "ATT",
               method = "score", data = df, alpha = 2)
summary(fits2m)

# Covariate balancing weighting function
fitcbm <- nawt(formula = formula_m, outcome = "y", estimand = "ATT",
               method = "cb", data = df)
summary(fitcbm)

# Standard logistic regression
fits0m <- nawt(formula = formula_m, outcome = "y", estimand = "ATT",
               method = "score", data = df, alpha = 0)
summary(fits0m)

# Empirical example
# Load the LaLonde data
data(LaLonde)
formula_l <- as.formula("exper ~ age + I(age^2) + educ + I(educ^2) +
                        black + hisp + married + nodegr +
                        I(re75 / 1000) + I(re75 == 0) + I(re74 / 1000)")

```

```

# Experimental benchmark
mean(subset(LaLonde, exper == 1 & treat == 1)$re78) -
  mean(subset(LaLonde, exper == 1 & treat == 0)$re78)

# Power weighting function with alpha = 2
fits2l <- nawt(formula = formula_1, estimand = "ATT", method = "score",
               outcome = "re78", data = LaLonde, alpha = 2)
mean(subset(LaLonde, exper == 1 & treat == 1)$re78) -
  with(LaLonde, sum((1 - exper) * re78 * fits2l$weights) /
        sum((1 - exper) * fits2l$weights))

# Covariate balancing weighting function
fitcbl <- nawt(formula = formula_1, estimand = "ATT", method = "cb",
               outcome = "re78", data = LaLonde)
mean(subset(LaLonde, exper == 1 & treat == 1)$re78) -
  with(LaLonde, sum((1 - exper) * re78 * fitcbl$weights) /
        sum((1 - exper) * fitcbl$weights))

# Standard logistic regression
fits0l <- nawt(formula = formula_1, estimand = "ATT", method = "score",
               outcome = "re78", data = LaLonde, alpha = 0)
mean(subset(LaLonde, exper == 1 & treat == 1)$re78) -
  with(LaLonde, sum((1 - exper) * re78 * fits0l$weights) /
        sum((1 - exper) * fits0l$weights))

```

plot.nawt

Plot a scattered plot comparing the navigated weighting and naive estimation

Description

Plots a scattered plot comparing the resulting inverse probability weights estimated by the navigated weighting and the standard logistic regression.

Usage

```
## S3 method for class 'nawt'
plot(x, ...)
```

Arguments

`x` an object of class “nawt”, usually, a result of a call to [nawt](#).
`...` additional arguments to be passed to `plot`.

Details

The x-axis shows the inverse probability weights estimated by estimating propensity scores with the standard logistic regression whereas the y-axis shows those with the navigated weighting. Excessively heavy weights on only a few observations in the navigated weighting may indicate the failure of the estimation.

Position of the legend is determined internally.

Value

No return value, called for side effects.

Author(s)

Hiroto Katsumata

See Also

[nawt](#), [plot](#)

Examples

```
# Simulation from Kang and Shafer (2007) and Imai and Ratkovic (2014)
tau <- 10
set.seed(12345)
n <- 1000
X <- matrix(rnorm(n * 4, mean = 0, sd = 1), nrow = n, ncol = 4)
prop <- 1 / (1 + exp(X[, 1] - 0.5 * X[, 2] + 0.25 * X[, 3] + 0.1 * X[, 4]))
treat <- rbinom(n, 1, prop)
y <- 210 + 27.4 * X[, 1] + 13.7 * X[, 2] + 13.7 * X[, 3] + 13.7 * X[, 4] +
  tau * treat + rnorm(n)

# Data frame and formulas for propensity score estimation
df <- data.frame(X, treat, y)
colnames(df) <- c("x1", "x2", "x3", "x4", "treat", "y")
formula_c <- as.formula(treat ~ x1 + x2 + x3 + x4)

# Power weighting function with alpha = 2
# ATT estimation
fitatt <- nawt(formula = formula_c, outcome = "y", estimand = "ATT",
  method = "score", data = df, alpha = 2)
plot(fitatt)

# ATE estimation
fiteate <- nawt(formula = formula_c, outcome = "y", estimand = "ATE",
  method = "score", data = df, alpha = 2)
plot(fiteate)
```

plot_omega

Plot weights for propensity score estimation in the navigated weighting

Description

Plots weight of each observation in the score condition $\omega(\pi)$ for propensity score estimation and estimated propensity score distribution in the navigated weighting.

Usage

```
plot_omega(object, relative = TRUE)
```

Arguments

object	an object of class “nawt”, usually, a result of a call to <code>nawt</code> . Note that it cannot be used when the object is a result of a call to <code>nawt</code> where <code>method = “both”</code> and <code>twostep = FALSE</code> .
relative	a logical value indicating whether or not relative weights standardized to have mean one are shown.

Details

The x-axis shows estimated propensity scores, and the y-axis shows weight of each observation in propensity score estimation. When `estimand = “ATE”`, the navigated weighting estimates two propensity scores for each observation; one for estimating the average of the potential outcomes with treatment and the other for estimating the average of the potential outcomes without treatment. Therefore, there are two weighting functions for estimating two sets of propensity scores and two propensity score distributions. Points rising to the right and a solid curve represent the weighting functions and distribution of propensity scores for estimating the average of the potential outcomes without treatment whereas points rising to the left and a dashed curve represent the weighting functions and distribution of propensity scores for estimating the average of the potential outcomes with treatment.

Position of the legend is determined internally.

Value

No return value, called for side effects.

Author(s)

Hiroto Katsumata

Examples

```
# Simulation from Kang and Shafer (2007) and Imai and Ratkovic (2014)
tau <- 10
set.seed(12345)
n <- 1000
X <- matrix(rnorm(n * 4, mean = 0, sd = 1), nrow = n, ncol = 4)
prop <- 1 / (1 + exp(X[, 1] - 0.5 * X[, 2] + 0.25 * X[, 3] + 0.1 * X[, 4]))
treat <- rbinom(n, 1, prop)
y <- 210 + 27.4 * X[, 1] + 13.7 * X[, 2] + 13.7 * X[, 3] + 13.7 * X[, 4] +
  tau * treat + rnorm(n)

# Data frame and formulas for propensity score estimation
df <- data.frame(X, treat, y)
colnames(df) <- c("x1", "x2", "x3", "x4", "treat", "y")
formula_c <- as.formula(treat ~ x1 + x2 + x3 + x4)
```

```

# Power weighting function with alpha = 2
# ATT estimation
fitatt <- nawt(formula = formula_c, outcome = "y", estimand = "ATT",
              method = "score", data = df, alpha = 2)
plot_omega(fitatt)

# ATE estimation
fitate <- nawt(formula = formula_c, outcome = "y", estimand = "ATE",
              method = "score", data = df, alpha = 2)
plot_omega(fitate)

# Use method = "both"
# Two-step estimation
fitateb2s <- nawt(formula = formula_c, outcome = "y", estimand = "ATE",
                 method = "both", data = df, alpha = 2, twostep = TRUE)
plot_omega(fitateb2s)

# Continuously-updating GMM estimation
## Not run:
fitatebco <- nawt(formula = formula_c, outcome = "y", estimand = "ATE",
                 method = "both", data = df, alpha = 2, twostep = FALSE)
plot_omega(fitatebco) # error
## End(Not run)

```

print.nawt

Print navigated weighting estimation

Description

Prints a fitted nawt object.

Usage

```
## S3 method for class 'nawt'
print(x, ...)
```

Arguments

x an object of class “nawt”, usually, a result of a call to [nawt](#).
... additional arguments to be passed to print.

Value

No retrun value, called for side effects.

Author(s)

Hiroto Katsumata

See Also[nawt](#), [print](#)

`summary.nawt`*Summarizing navigated weighting estimation*

Description

Prints a summary of a fitted nawt object.

Usage

```
## S3 method for class 'nawt'
summary(object, ...)
```

Arguments

`object` an object of class “nawt”, usually, a result of a call to [nawt](#).
`...` additional arguments to be passed to `summary`.

Details

Prints a summary of a nawt object, in a format similar to `glm`.

Value

`call` the matched call.
`est` the point estimate of the parameter of interest.
`coefficients` a table including coefficients, standard errors, z-values, and two-sided p-values.
`effN_ps` the effective sample size for the propensity score estimation.
`effN_est` the effective sample size for the parameter of interest estimation.

Author(s)

Hiroto Katsumata

See Also[nawt](#), [summary](#)**Examples**

```
# For examples see example(nawt)
```

Index

* **datasets**

LaLonde, [4](#)

cbcheck, [2](#), [7](#)

formula, [6](#)

LaLonde, [4](#), [5](#)

nawt, [2](#), [5](#), [10–14](#)

plot, [11](#)

plot.nawt, [10](#)

plot_omega, [11](#)

print, [14](#)

print.nawt, [13](#)

summary, [14](#)

summary.nawt, [7](#), [8](#), [14](#)