

# Package ‘VAR.etc’

May 7, 2026

**Type** Package

**Title** VAR Modelling: Estimation, Testing, and Prediction

**Version** 1.1

**Date** 2023-08-31

**Author** Jae. H. Kim

**Maintainer** Jae H. Kim <jaekim8080@gmail.com>

**Description** A collection of the functions for estimation, hypothesis testing, prediction for stationary vector autoregressive models.

**License** GPL-2

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2023-08-31 08:40:02 UTC

## Contents

VAR.etc-package . . . . .	2
dat . . . . .	2
data1 . . . . .	3
PR.Fore . . . . .	3
PR.IARM . . . . .	4
PR.order . . . . .	5
Rmatrix . . . . .	6
VAR.BaBPR . . . . .	7
VAR.Boot . . . . .	8
VAR.BPR . . . . .	9
VAR.est . . . . .	10
VAR.FOR . . . . .	11
VAR.Fore . . . . .	12
VAR.irf . . . . .	13
VAR.LR . . . . .	14
VAR.Pope . . . . .	16
VAR.Rest . . . . .	17
VAR.select . . . . .	18
VAR.Wald . . . . .	19

**Index****21**

---

VAR.etp-package	<i>VAR Modelling: Estimation, Testing, and Prediction</i>
-----------------	---

---

**Description**

Estimation, Hypothesis Testing, Prediction in Stationary Vector Autoregressive Models

**Details**

Package: VAR.etp  
Type: Package  
Version: 1.1  
Date: 2023-08-31  
License: GPL-2

The data set dat.rda is from Lutkepohl's book.

It is German Macrodata in log difference.

Bootstrap bias-correction and prediction intervals are also included.

Estimation and Forecasting based on Predictive Regression is also included.

**Author(s)**

Jae H. Kim

Maintainer: Jae H. Kim

---

dat	<i>German investment income consumption in log difference</i>
-----	---

---

**Description**

Lutkepohl's data

**Usage**

data(dat)

**References**

Lutkepohl, H. 2005, *New Introduction to Multiple Time Series Analysis*, Springer

**Examples**

data(dat)

---

data1	<i>stock return data used in Kim (2014)</i>
-------	---

---

**Description**

stock return data used in Kim (2014)

**Usage**

```
data(data1)
```

**References**

Kim, J.H. 2014, Testing for parameter restrictions in a stationary VAR model: a bootstrap alternative. *Economic Modelling*, 41, 267-273.

**Examples**

```
data(data1)
```

---

PR.Fore	<i>Improved Augmented Regression Method for Predictive Regression</i>
---------	---

---

**Description**

Function for forecasting based on Improved ARM

**Usage**

```
PR.Fore(x, y, M, h = 10)
```

**Arguments**

x	predictor or matrix of predictors in column
y	variable to be predicted, usually stock return
M	Estimation results of the function PR.IARM
h	forecasting period

**Details**

Function for forecasting based on Improved ARM

**Value**

Fore	Out-of sample and dynamic forecasts for y and x
------	---

**Note**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance 25, 13-15.

**Author(s)**

jae H. Kim

**References**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance 25, 13-15.

**Examples**

```
data(data1)
# Replicating Table 5 (excess return)
y=data1$ret.nyse.vw*100 -data1$tbill*100
x=cbind(log(data1$dy.nyse), data1$tbill*100); k=ncol(x)
p=4
Rmat1=Rmatrix(p,k,type=1,index=1); Rmat=Rmat1$Rmat; rvec=Rmat1$rvec
M=PR.IARM(x,y,p,Rmat,rvec)
PRF=PR.Fore(x,y,M)
```

---

PR.IARM

*Improved Augmented Regression Method (IARM) for Predictive Regression*

---

**Description**

Function for Improved ARM (IARM) estimation and testing

**Usage**

```
PR.IARM(x, y, p, Rmat = diag(k * p), rvec = matrix(0, nrow = k * p))
```

**Arguments**

x	predictor or a matrix of predictors in column
y	variable to be predicted, usually data1 return
p	AR order
Rmat	Restriction matrix, refer to function Rmatrix
rvec	Restriction matrix, refer to function Rmatrix

**Details**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance, 26, 13-25.

**Value**

LS	Ordinary Least Squares Estimators
IARM	IARM Estimators
AR	AR parameter estimators
ARc	Bias-corrected AR parameter estimators
Fstats	Fstats and their p-values
Covbc	Covariance matrix of the IARM estimators (for the predictive coefficients only)

**Note**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance, 26, 13-25.

**Author(s)**

Jae H. Kim

**References**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance, 26, 13-25.

**Examples**

```
data(data1)
# Replicating Table 5 (excess return) of Kim (2014)
y=data1$ret.nyse.vw*100 -data1$tbill*100
x=cbind(log(data1$dy.nyse), data1$tbill*100);

Rmat1=Rmatrix(p=1,k=2,type=1,index=0); Rmat=Rmat1$Rmat; rvec=Rmat1$rvec
M=PR.IARM(x,y,p=1,Rmat,rvec)
```

---

PR.order

*Improved Augmented Regression Method for Predictive Regression*

---

**Description**

Function to select the order p by AIC or BIC

**Usage**

```
PR.order(x, y, pmax = 10)
```

**Arguments**

x	predictor or a matrix of predictors in column
y	variable to be predicted, usually stock return
pmax	maximum order for order selection

**Details**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance 25, 13-15.

**Value**

p.aic            order chosen by AIC  
p.aic            order chosen by BIC

**Note**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance 25, 13-15.

**Author(s)**

Jae H. Kim

**References**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance 25, 13-15.

**Examples**

```
data(data1)
# Replicating Table 5 (excess return)
y=data1$ret.nyse.vw*100 -data1$tbill*100
x=cbind(log(data1$dy.nyse), data1$tbill*100); k=ncol(x)

p=PR.order(x,y,pmax=10)$p.bic; # AR(1)
```

---

Rmatrix

*Improved Augmented Regression Method for Predictive Regression*

---

**Description**

Function to generate restriction matrices

**Usage**

```
Rmatrix(p, k, type = 1, index = 0)
```

**Arguments**

p            AR order  
k            number of predictors  
type        type = 1: H0: b1=b2=b3=0; type = 2: H0: b1+b2+b3=0  
index       index=0 : H0 applies for all parameters; index=k : H0 applies for kth predictor

**Details**

Function to generate restriction matrices

**Value**

Rmat            this value should be passed to PR.IARM  
 rvec            this value should be passed to PR.IARM

**Author(s)**

Jae H. Kim

**References**

Kim J.H., 2014, Predictive Regression: Improved Augmented Regression Method, Journal of Empirical Finance 25, 13-15.

**Examples**

```
Rmat1=Rmatrix(p=1,k=1,type=2,index=1); Rmat=Rmat1$Rmat; rvec=Rmat1$rvec
```

---

 VAR.BaBPR

---

*Bootstrap-after-Bootstrap Prediction Intervals for VAR(p) Model*


---

**Description**

Bias-correction given with stationarity Correction

**Usage**

```
VAR.BaBPR(x, p, h, nboot = 500, nb = 200, type = "const", alpha = 0.95)
```

**Arguments**

x                data matrix in column  
 p                AR order  
 h                forecasting period  
 nboot            number of 2nd-stage bootstrap iterations  
 nb                number of 1st-stage bootstrap iterations  
 type             "const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend  
 alpha            100(1-alpha) percent prediction intervals

**Details**

Bias-correction given with stationarity Correction

**Value**

Intervals	Prediction Intervals
Forecast	Point Forecasts
alpha	Probability Content of Prediction Intervals

**Note**

Bias-correction given with stationarity Correction

**Author(s)**

Jae H. Kim

**References**

Kim, J. H. (2001). Bootstrap-after-bootstrap prediction intervals for autoregressive models, *Journal of Business & Economic Statistics*, 19, 117-128.

**Examples**

```
data(dat)
VAR.BaBPR(dat,p=2,h=10,nboot=200,nb=100,type="const",alpha=0.95)
# nboot and nb are set to low numbers for fast execution in the example
# In actual implementation, use higher numbers such as nboot=1000, nb=200
```

---

VAR.Boot

*Bootstrapping VAR(p) model: bias-correction based on the bootstrap*

---

**Description**

The function returns bias-corrected parameter estimators and Bias estimators based on the bootstrap

**Usage**

```
VAR.Boot(x, p, nb = 200, type = "const")
```

**Arguments**

x	data matrix in column
p	AR order
nb	number of bootstrap iterations
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend

**Details**

Kilian's (1998) stationarity-correction is used for bias-correction

**Value**

coef	coefficient matrix
resid	matrix of residuals
sigu	residual covariance matrix
Bias	Bootstrap Bias Estimator

**Author(s)**

Jae H. Kim

**References**

Kilian, L. (1998). Small sample confidence intervals for impulse response functions, *The Review of Economics and Statistics*, 80, 218 - 230.

**Examples**

```
data(dat)
VAR.Boot(dat, p=2, nb=200, type="const")
```

---

 VAR.BPR

---

*Bootstrap Prediction Intervals for VAR(p) Model*


---

**Description**

No Bias-correction is given

**Usage**

```
VAR.BPR(x, p, h, nboot = 500, type = "const", alpha = 0.95)
```

**Arguments**

x	data matrix in column
p	AR order
h	forecasting period
nboot	number of bootstrap iterations
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend
alpha	100(1-alpha) percent prediction intervals

**Details**

Bootstrap Prediction Intervals for VAR(p) Model

**Value**

Intervals	Prediction Intervals
Forecast	Point Forecasts
alpha	Probability Content of Prediction Intervals

**Note**

No Bias-correction is given

**Author(s)**

Jae H. Kim

**References**

Kim, J. H. (2001). Bootstrap-after-bootstrap prediction intervals for autoregressive models, *Journal of Business & Economic Statistics*, 19, 117-128.

**Examples**

```
data(dat)
VAR.BPR(dat,p=2,h=10,nboot=200,type="const",alpha=0.95)
# nboot is set to a low number for fast execution in the example
# In actual implementation, use higher number such as nboot=1000
```

---

VAR.est

*Estimation of unrestricted VAR(p) model parameters*

---

**Description**

This function returns least-squares estimation results for VAR(p) model

**Usage**

```
VAR.est(x, p, type = "const")
```

**Arguments**

x	data matrix in column
p	AR order
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend

**Details**

VAR estimation

**Value**

coef	coefficient matrix
resid	matrix of residuals
sigu	residual covariance matrix
zzmat	data moment matrix
zmat	data moment matrix
tratio	matrix of ratio corresponding to coef matrix

**Note**

See Chapter 3 of Lutkepohl (2005)

**Author(s)**

Jae H. Kim

**References**

Lutkepohl, H. 2005, New Introduction to Multiple Time Series Analysis, Springer

**Examples**

```
#replicating Section 3.2.3 of of Lutkepohl (2005)
data(dat)
M=VAR.est(dat,p=2,type="const")
print(M$coef)
print(M$tratio)
```

---

VAR.FOR

*VAR Forecasting*

---

**Description**

Generate point forecasts and prediction intervals

**Usage**

```
VAR.FOR(x, p, h, type = "const", alpha = 0.95)
```

**Arguments**

x	data matrix in column
p	VAR order
h	Forecasting Periods
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend
alpha	100(1-alpha) percent prediction intervals

**Details**

Prediction Intervals are based on normal approximation

**Value**

Intervals	Prediction Intervals, out-of-sample and dynamic
Forecast	Point Forecasts, out-of-sample and dynamic
alpha	Probability Content of Prediction Intervals

**Note**

See Chapter 3 of Lutkepohl (2005)

**Author(s)**

Jae H. Kim

**References**

Lutkepohl, H. 2005, New Introduction to Multiple Time Series Analysis, Springer

**Examples**

```
#replicating Section 3.5.3 of Lutkepohl (2005)
data(dat)
VAR.FOR(dat,p=2,h=10,type="const",alpha=0.95)
```

---

VAR.Fore

*VAR Forecasting*


---

**Description**

Generate point forecasts using the estimated VAR coefficient matrix

**Usage**

```
VAR.Fore(x, b, p, h, type = "const")
```

**Arguments**

x	data matrix in column
b	matrix of coefficients from VAR.est or VAR.Rest
p	VAR order
h	Forecasting Periods
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend

**Details**

Generate point forecasts using the estimated VAR coefficient matrix

**Value**

Fore Point Forecasts, out-of-sample and dynamic

**Note**

See Chapter 3 of Lutkepohl (2005)

**Author(s)**

Jae H. Kim

**References**

Lutkepohl, H. 2005, New Introduction to Multiple Time Series Analysis, Springer

**Examples**

```
#replicating Section 3.5.3 of Lutkepohl (2005)
data(dat)
b=VAR.est(dat,p=2,type="const")$coef
VAR.Fore(dat,b,p=2,h=10,type="const")
```

---

VAR.irf	<i>Orthogonalized impluse response functions from an estimated VAR(p) model</i>
---------	---

---

**Description**

This function returns Orthogonalized impluse response functions

**Usage**

```
VAR.irf(b, p, sigu, h=10,graphs=FALSE)
```

**Arguments**

b	VAR coefficient matrix, from VAR.est or similar estimation function
p	VAR order
sigu	VAR residual covariance matrix, from VAR.est or similar estimation function
h	response horizon, the default is set to 10
graphs	logical, if TRUE, show the impulse-response functions, the default is FALSE

**Details**

VAR impulse response functions

**Value**

impat            matrix that contains orthogonalized impulse-responses

**Note**

See Lutkepohl (2005) for details

**Author(s)**

Jae H. Kim

**References**

Lutkepohl, H. 2005, New Introduction to Multiple Time Series Analysis, Springer

**Examples**

```
#replicating Table 3.4 and Figure 3.11 Lutkepohl (2005)
data(dat)
M=VAR.est(dat,p=2,type="const")
b=M$coef; sigu=M$sigu
VAR.irf(b,p=2,sigu,graphs=TRUE)
```

---

VAR.LR

*The Likelihood Ratio test for parameter restrictions*

---

**Description**

Likelihood Ratio test for zero parameter restrictions based on system VAR estimation

Bootstrap option is available: iid bootstrap or wild bootstrap

Bootstrap is conducted under the null hypothesis using estimated GLS estimation: see Kim (2014)

**Usage**

```
VAR.LR(x, p, restrict0, restrict1, type = "const",bootstrap=0,nb=500)
```

**Arguments**

x	data matrix in column
p	VAR order
restrict0	Restriction matrix under H0
restrict1	Restriction matrix under H1, if "full", the full VAR is estimated under H1
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend
bootstrap	0 for no bootstrap; 1 for iid bootstrap; 2 for wild bootstrap
nb	the number of bootstrap iterations

**Details**

Restriction matrix is of  $m$  by 3 matrix where  $m$  is the number of restrictions. A typical row of this matrix  $(k,i,j)$ , which means that  $(i,j)$  element of  $A_k$  matrix is set to 0.  $A_k$  is a VAR coefficient matrix ( $k = 1, \dots, p$ ).

The bootstrap test is conducted using the GLS estimation under the parameter restrictions implied by the null hypothesis: see Kim (2014) for details.

Kim (2014) found that the bootstrap based on OLS can show inferior small sample properties.

There are two versions of the bootstrap: the first is based on the iid resampling and the second based on wild bootstrapping.

The Wild bootstrap is conducted with Mammen's two-point distribution.

**Value**

LRstat	LR test statistic
pval	p-value of the LR test
Boot.pval	p-value of the test based on bootstrapping

**Note**

See Chapter 4 of Lutkepohl (2005)

**Author(s)**

Jae H. Kim

**References**

- Lutkepohl, H. 2005, *New Introduction to Multiple Time Series Analysis*, Springer
- Kim, J.H. 2014, Testing for parameter restrictions in a stationary VAR model: a bootstrap alternative. *Economic Modelling*, 41, 267-273.

**Examples**

```

data(dat)
#replicating Table 4.4 of Lutkepohl (2005)
restrict1="full";
restrict0 = rbind(c(4,1,1), c(4,1,2), c(4,1,3), c(4,2,1),
c(4,2,2),c(4,2,3),c(4,3,1),c(4,3,2),c(4,3,3))
VAR.LR(dat,p=4,restrict0,restrict1,type="const")

```

VAR.Pope

*Bias-correction for VAR parameter estimators based on Pope's formula*

**Description**

The function returns bias-corrected parameter estimators and Bias estimators based on Pope's asymptotic formula

**Usage**

```
VAR.Pope(x, p, type = "const")
```

**Arguments**

x	data matrix in column
p	AR order
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend

**Details**

Kilian's (1998) stationarity-correction is used for bias-correction

**Value**

coef	Bias-corrected coefficient matrix
resid	matrix of residuals
sigu	residual covariance matrix
Bias	Bias Estimate

**Author(s)**

Jae H. Kim

## References

- Kim, J. H. 2004, Bias-corrected bootstrap prediction regions for Vector Autoregression, Journal of Forecasting 23, 141-154.
- Kilian, L. (1998). Small sample confidence intervals for impulse response functions, The Review of Economics and Statistics, 80, 218 - 230.
- Nicholls DF, Pope AL. 1988, Bias in estimation of multivariate autoregression. Australian Journal of Statistics, 30A, 296-309.
- Pope AL. 1990. Biases of estimators in multivariate non-Gaussian autoregression, Journal of Time Series Analysis 11, 249-258.

## Examples

```
data(dat)
VAR.Pope(dat, p=2, type="const")
```

---

 VAR.Rest

---

*VAR parameter estimation with parameter restrictions*


---

## Description

Estimation of VAR with 0 restrictions on parameters

## Usage

```
VAR.Rest(x, p, restrict, type = "const", method = "gls")
```

## Arguments

x	data matrix in column
p	VAR order
restrict	Restriction matrix under H0
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend
method	"ols" for OLS estimation, "gls" for EGLS estimation

## Details

Restriction matrix is of m by 3 matrix where m is the number of restrictions. A typical row of this matrix (k,i,j), which means that (i,j) element of  $A_k$  matrix is set to 0.  $A_k$  is a VAR coefficient matrix ( $k = 1, \dots, p$ ).

**Value**

coef	coefficient matrix
resid	matrix of residuals
sigu	residual covariance matrix
zmat	data matrix
tstat	matrix of t-ratio corresponding to coef matrix

**Note**

See Chapter 5 of Lutkepohl

**Author(s)**

Jae H. Kim

**References**

Lutkepohl, H. 2005, New Introduction to Multiple Time Series Analysis, Springer

**Examples**

```
data(dat)
#replicating Section 5.2.10 of Lutkepohl (2005)
restrict = rbind( c(1,1,2),c(1,1,3),c(1,2,1),c(1,2,2), c(1,3,1),
c(2,1,1), c(2,1,2),c(2,1,3), c(2,2,2), c(2,2,3),c(2,3,1), c(2,3,3),
c(3,1,1), c(3,1,2), c(3,1,3), c(3,2,1), c(3,2,2), c(3,2,3), c(3,3,1),c(3,3,3),
c(4,1,2), c(4,1,3), c(4,2,1), c(4,2,2), c(4,2,3), c(4,3,1),c(4,3,2),c(4,3,3))
M= VAR.Rest(dat,p=4,restrict,type="const",method="gls")
print(M$coef)
print(M$tstat)
```

---

VAR.select

*Order Selection for VAR models*


---

**Description**

AIC, HQ, or SC can be used

**Usage**

```
VAR.select(x, type = "const", ic = "aic", pmax)
```

**Arguments**

x	data matrix in column
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend
ic	choose one of "aic", "sc", "hq"
pmax	the maximum VAR order

**Details**

Order Section Criterion

**Value**

IC	Values of information criterion for VAR models
p	AR order selected

**Note**

See Chapter 4 of Lutkepohl

**Author(s)**

JAe H. Kim

**References**

Lutkepohl, H. 2005, *New Introduction to Multiple Time Series Analysis*, Springer

**Examples**

```
data(dat)
#replicating Section 4.3.1 of Lutkepohl (2005)
VAR.select(dat,pmax=4,ic="aic")
```

---

VAR.Wald

*Wald test for parameter restrictions*

---

**Description**

Wald test for zero parameter restrictions based on system VAR estimation

Bootstrap option is available: iid bootstrap or wild bootstrap

Bootstrap is conducted under the null hypothesis using estimated GLS estimation: see Kim (2014)

**Usage**

```
VAR.Wald(x, p, restrict, type = "const",bootstrap=0,nb=500)
```

**Arguments**

x	data matrix in column
p	VAR order
restrict	Restriction matrix under H0
type	"const" for the AR model with intercept only, "const+trend" for the AR model with intercept and trend
bootstrap	0 for no bootstrap; 1 for iid bootstrap; 2 for wild bootstrap
nb	the number of bootstrap iterations

**Details**

Restriction matrix is of m by 3 matrix where m is the number of restrictions. A typical row of this matrix (k,i,j), which means that (i,j) element of Ak matrix is set to 0. Ak is a VAR coefficient matrix (k = 1,...,p). Under H1, the model is full VAR.

The bootstrap test is conducted using the GLS estimation under the parameter restrictions implied by the null hypothesis: see Kim (2014) for details.

Kim (2014) found that the bootstrap based on OLS can show inferior small sample properties.

There are two versions of the bootstrap: the first is based on the iid resampling and the second based on wild bootstrapping.

The Wild bootstrap is conducted with Mammen's two-point distribution.

**Value**

Fstat	Wald test statistic
pval	p-value of the test based on F-distribution
Boot.pval	p-value of the test based on bootstrapping

**Note**

See Chapter 3 of Lutkepohl

**Author(s)**

Jae H. Kim

**References**

Lutkepohl, H. 2005, New Introduction to Multiple Time Series Analysis, Springer.

Kim, J.H. 2014, Testing for parameter restrictions in a stationary VAR model: a bootstrap alternative. *Economic Modelling*, 41, 267-273.

**Examples**

```
data(dat)
#replicating Section 3.6.2 of Lutkepohl (2005)
restrict = rbind( c(1,1,2),c(1,1,3), c(2,1,2),c(2,1,3))
VAR.Wald(dat,p=2,restrict,type="const")
```

# Index

- \* **package**
  - VAR.etp-package, 2
- \* **ts**
  - dat, 2
  - data1, 3
  - PR.IARM, 4
  - PR.order, 5
  - Rmatrix, 6
  - VAR.BaBPR, 7
  - VAR.Boot, 8
  - VAR.BPR, 9
  - VAR.est, 10
  - VAR.FOR, 11
  - VAR.Fore, 12
  - VAR.irf, 13
  - VAR.LR, 14
  - VAR.Pope, 16
  - VAR.Rest, 17
  - VAR.select, 18
  - VAR.Wald, 19
- \* **ys**
  - PR.Fore, 3
  
- dat, 2
- data1, 3
  
- PR.Fore, 3
- PR.IARM, 4
- PR.order, 5
  
- Rmatrix, 6
  
- VAR.BaBPR, 7
- VAR.Boot, 8
- VAR.BPR, 9
- VAR.est, 10
- VAR.etp (VAR.etp-package), 2
- VAR.etp-package, 2
- VAR.FOR, 11
- VAR.Fore, 12
- VAR.irf, 13
- VAR.LR, 14
- VAR.Pope, 16
- VAR.Rest, 17
- VAR.select, 18
- VAR.Wald, 19