

# ***Integration by differentiation***

## ***Doing modern statistics in AD Model Builder***

Hans J. Skaug

`skaug@imr.no`

Institute of Marine Research

Bergen, Norway

# *A programming language perspective*

- Standard AD Model Builder (based on C++):

Declaration

Functionality

---

```
vector t(10);
```

```
obj_function f;
```

$$\hat{t} = \operatorname{argmax}_t f(t)$$

# *A programming language perspective*

- Standard AD Model Builder (based on C++):

Declaration

Functionality

---

```
vector t(10);
```

```
obj_function f;
```

$$\hat{t} = \operatorname{argmax}_t f(t)$$

- New feature:

```
random_eff u(500);
```

$$\hat{t} = \operatorname{argmax}_t \int f(t, \mathbf{u}) d\mathbf{u}$$

# *Integration: Why and how?*

- Why?
  - Statistics: Random effects, Bayesian models, state space models...
  - Make technical details transparent

# *Integration: Why and how?*

- Why?
  - Statistics: Random effects, Bayesian models, state space models...
  - Make technical details transparent
- How?
  - Third order derivatives by AD
  - 1 forward → 2 reverse sweeps
  - Operator overloading in C++

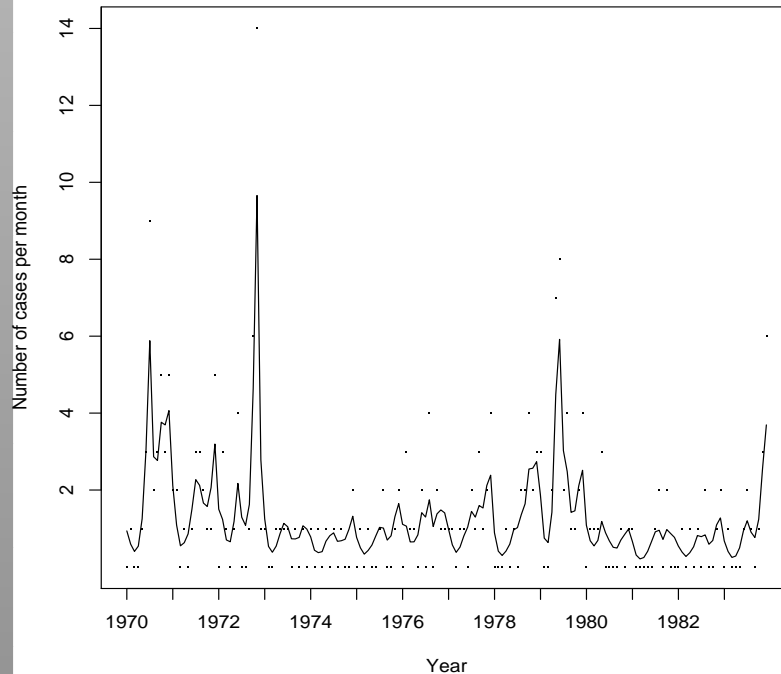
# *Why hasn't AD made it in statics, yet?*

- Statistics is more than:
  - The method of least squares
  - Non-linear regression

# *Why hasn't AD made it in statics, yet?*

- Statistics is more than:
  - The method of least squares
  - Non-linear regression
- Statisticians deal with:
  - Discrete measurements: dead/alive
  - much, much more ...

# Discrete valued time series



- Monthly numbers of Polio cases in the US
- Gaussian distribution inappropriate!
- Autocorrelated counts

- Poisson distribution:

$$f(y_i | \lambda_i) = \frac{\lambda_i^{y_i}}{y_i!} \exp(-\lambda_i)$$

# *Nonparametric smoothing*

- The log-likelihood:

$$\sum_{i=1}^{168} \log \{f(y_i | \lambda_i)\}$$

# Nonparametric smoothing

- The log-likelihood:

$$\sum_{i=1}^{168} \log \{f(y_i | \lambda_i)\}$$

- Penalized log-likelihood:

$$\sum_{i=1}^{168} \log \{f(y_i | \lambda_i)\} - \theta \sum_{i=2}^{168} (\log \lambda_i - \log \lambda_{i-1})^2$$

# Nonparametric smoothing

- The log-likelihood:

$$\sum_{i=1}^{168} \log \{f(y_i | \lambda_i)\}$$

- Penalized log-likelihood:

$$\sum_{i=1}^{168} \log \{f(y_i | \lambda_i)\} - \theta \sum_{i=2}^{168} (\log \lambda_i - \log \lambda_{i-1})^2$$

- How to determine  $\theta$ ?

## *Side-step: Regularization*

- Minimize w.r.t.  $\mathbf{u}$ :

$$(\mathbf{y} - \mathbf{X}\mathbf{u})' (\mathbf{y} - \mathbf{X}\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

## *Side-step: Regularization*

- Minimize w.r.t.  $\mathbf{u}$ :

$$(\mathbf{y} - \mathbf{X}\mathbf{u})' (\mathbf{y} - \mathbf{X}\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

- A more general objective function

$$-f_{\theta}(\mathbf{y}|\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

## *Side-step: Regularization*

- Minimize w.r.t.  $\mathbf{u}$ :

$$(\mathbf{y} - \mathbf{X}\mathbf{u})' (\mathbf{y} - \mathbf{X}\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

- A more general objective function

$$-f_{\theta}(\mathbf{y}|\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

- Cannot minimize jointly w.r.t.  $\mathbf{u}$  and  $\theta$

## *Side-step: Regularization*

- Minimize w.r.t.  $\mathbf{u}$ :

$$(\mathbf{y} - \mathbf{X}\mathbf{u})' (\mathbf{y} - \mathbf{X}\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

- A more general objective function

$$-f_{\theta}(\mathbf{y}|\mathbf{u}) + \mathbf{u}' K_{\theta} \mathbf{u}$$

- Cannot minimize jointly w.r.t.  $\mathbf{u}$  and  $\theta$
- Choosing the smoothing parameter  $\theta$ :
  - Cross-validation
  - **Maximum likelihood** (G. Wahba and others)

# Back to the polio data

- Penalty term ( $u_i = \log \lambda_i$ ):

$$\sigma^{-2} \sum_{i=2}^{168} (u_i - \rho u_{i-1})^2 = \mathbf{u}' K \mathbf{u}$$

$$K(\rho, \sigma) = \sigma^{-2} \begin{bmatrix} 1 + \rho^2 & -\rho & & & & \\ -\rho & 1 + \rho^2 & -\rho & & & \\ & -\rho & \ddots & \ddots & & \\ & & \ddots & \ddots & -\rho & \\ & & & -\rho & 1 + \rho^2 & \end{bmatrix}$$

# *Maximum likelihood estimation of $\theta$*

- "Prior" probability density of  $\mathbf{u}$

$$h_{\theta}(\mathbf{u}) \propto \det K_{\theta}^{1/2} \exp \{ -\mathbf{u}' K_{\theta} \mathbf{u} \}$$

# Maximum likelihood estimation of $\theta$

- "Prior" probability density of  $\mathbf{u}$

$$h_{\theta}(\mathbf{u}) \propto \det K_{\theta}^{1/2} \exp \{ -\mathbf{u}' K_{\theta} \mathbf{u} \}$$

- Estimation:

$$\hat{\theta} = \operatorname{argmax}_{\theta} \int f_{\theta}(\mathbf{y}|\mathbf{u}) h_{\theta}(\mathbf{u}) d\mathbf{u}.$$

# Maximum likelihood estimation of $\theta$

- "Prior" probability density of  $\mathbf{u}$

$$h_{\theta}(\mathbf{u}) \propto \det K_{\theta}^{1/2} \exp \{ -\mathbf{u}' K_{\theta} \mathbf{u} \}$$

- Estimation:

$$\hat{\theta} = \operatorname{argmax}_{\theta} \int f_{\theta}(\mathbf{y}|\mathbf{u}) h_{\theta}(\mathbf{u}) d\mathbf{u}.$$

- This establishes the need for integration!

# *The Laplace approximation*

- Penalized log-likelihood

$$g(\mathbf{y}, \mathbf{u}, \theta) = \log [f_{\theta}(\mathbf{y}|\mathbf{u})] + \log [h_{\theta}(\mathbf{u})]$$

# *The Laplace approximation*

- Penalized log-likelihood

$$g(\mathbf{y}, \mathbf{u}, \theta) = \log [f_{\theta}(\mathbf{y}|\mathbf{u})] + \log [h_{\theta}(\mathbf{u})]$$

- Taylor expansion around

$$\hat{\mathbf{u}}(\theta) = \underset{\mathbf{u}}{\operatorname{argmax}} g(\mathbf{y}, \mathbf{u}, \theta)$$

# The Laplace approximation

- Penalized log-likelihood

$$g(\mathbf{y}, \mathbf{u}, \theta) = \log [f_{\theta}(\mathbf{y}|\mathbf{u})] + \log [h_{\theta}(\mathbf{u})]$$

- Taylor expansion around

$$\hat{\mathbf{u}}(\theta) = \operatorname{argmax}_{\mathbf{u}} g(\mathbf{y}, \mathbf{u}, \theta)$$

- The Laplace approximation

$$L^*(\theta) = \det \{\mathbf{H}(\theta)\}^{-1/2} f_{\theta}(\mathbf{y}|\hat{\mathbf{u}}(\theta)) h_{\theta}(\hat{\mathbf{u}}(\theta))$$

where  $\mathbf{H}(\theta) = -\frac{\partial^2}{\partial \mathbf{u}^2} g(\mathbf{y}, \mathbf{u}, \theta) \Big|_{\mathbf{u}=\hat{\mathbf{u}}(\theta)}$

# ***Evaluation***

- The user writes the code for  $g$

# Evaluation

- The user writes the code for  $g$
- The Laplace approximation

$$L^*(\theta) = \det \{ \mathbf{H}(\theta) \}^{-1/2} \exp [g \{ \mathbf{y}, \hat{\mathbf{u}}(\theta), \theta \} ],$$

# Evaluation

- The user writes the code for  $g$
- The Laplace approximation

$$L^*(\theta) = \det \{ \mathbf{H}(\theta) \}^{-1/2} \exp [g \{ \mathbf{y}, \hat{\mathbf{u}}(\theta), \theta \} ],$$

- By numerical optimization:

$$\hat{\mathbf{u}}(\theta) = \underset{\mathbf{u}}{\operatorname{argmax}} g(\mathbf{y}, \mathbf{u}, \theta)$$

# Evaluation

- The user writes the code for  $g$
- The Laplace approximation

$$L^*(\theta) = \det \{ \mathbf{H}(\theta) \}^{-1/2} \exp [g \{ \mathbf{y}, \hat{\mathbf{u}}(\theta), \theta \} ],$$

- By numerical optimization:

$$\hat{\mathbf{u}}(\theta) = \underset{\mathbf{u}}{\operatorname{argmax}} g(\mathbf{y}, \mathbf{u}, \theta)$$

- By AD:  $\mathbf{H}(\theta) = \frac{\partial^2}{\partial \mathbf{u}^2} g(\mathbf{y}, \mathbf{u}, \theta) \Big|_{\mathbf{u}=\hat{\mathbf{u}}(\theta)}$

# Evaluation

- The user writes the code for  $g$
- The Laplace approximation

$$L^*(\theta) = \det \{ \mathbf{H}(\theta) \}^{-1/2} \exp [g \{ \mathbf{y}, \hat{\mathbf{u}}(\theta), \theta \}],$$

- By numerical optimization:

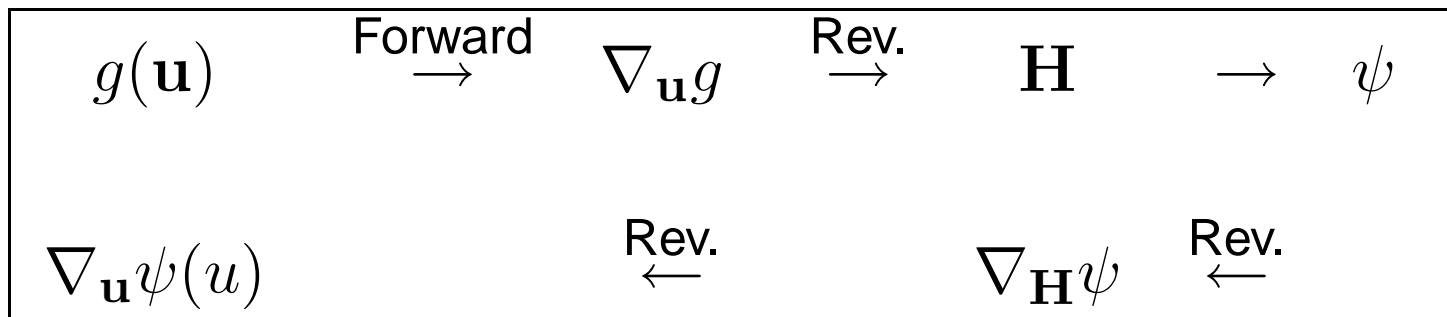
$$\hat{\mathbf{u}}(\theta) = \underset{\mathbf{u}}{\operatorname{argmax}} g(\mathbf{y}, \mathbf{u}, \theta)$$

- By AD:  $\mathbf{H}(\theta) = \frac{\partial^2}{\partial \mathbf{u}^2} g(\mathbf{y}, \mathbf{u}, \theta) \Big|_{\mathbf{u}=\hat{\mathbf{u}}(\theta)}$
- Need  $\nabla_{\theta} L^*(\theta)$  for optimization

# 1 forward $\rightarrow$ 2 reverse sweeps

- Find  $\nabla_{\mathbf{u}}\psi(u)$ , where

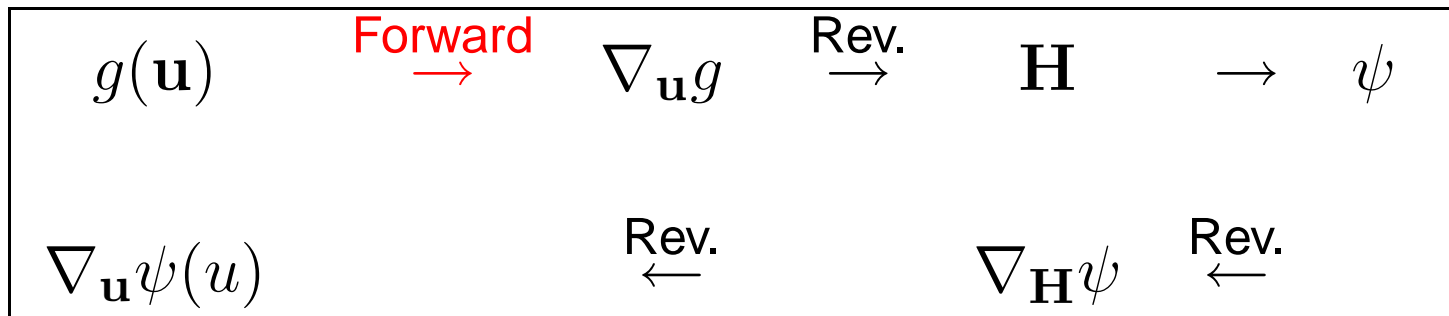
$$\begin{aligned}\psi(\mathbf{u}) &= \det \{\mathbf{H}(\mathbf{u})\} \\ \mathbf{H}(\mathbf{u}) &= \nabla_{\mathbf{u}\mathbf{u}}g(\mathbf{u})\end{aligned}$$



# 1 forward $\rightarrow$ 2 reverse sweeps

- Find  $\nabla_{\mathbf{u}}\psi(u)$ , where

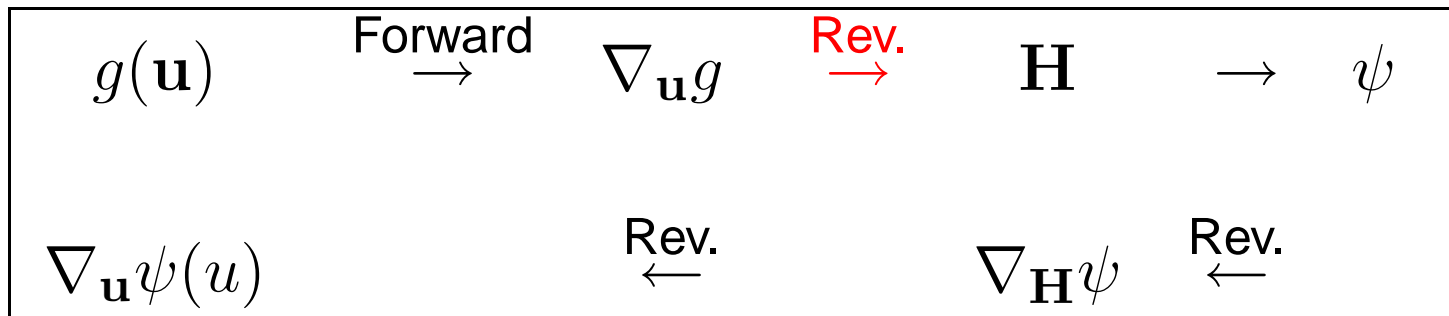
$$\begin{aligned}\psi(\mathbf{u}) &= \det \{\mathbf{H}(\mathbf{u})\} \\ \mathbf{H}(\mathbf{u}) &= \nabla_{\mathbf{u}\mathbf{u}}g(\mathbf{u})\end{aligned}$$



# 1 forward $\rightarrow$ 2 reverse sweeps

- Find  $\nabla_{\mathbf{u}}\psi(u)$ , where

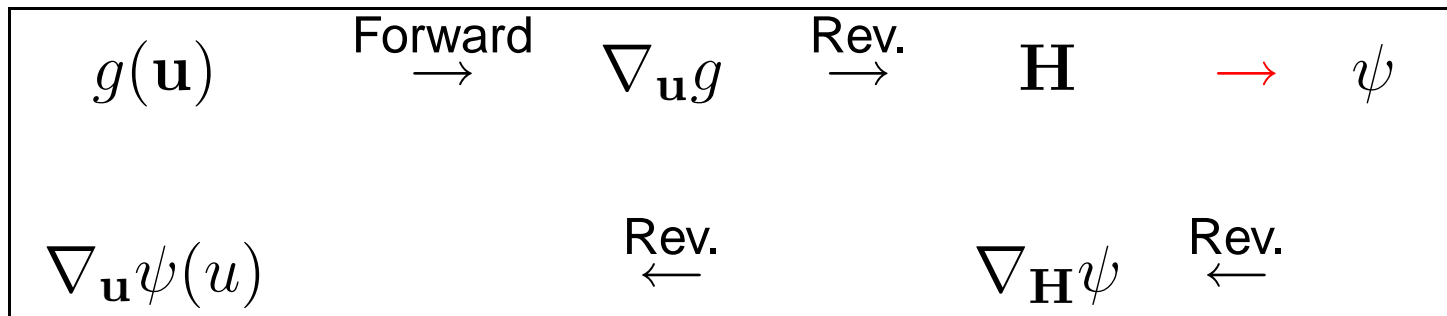
$$\begin{aligned}\psi(\mathbf{u}) &= \det \{\mathbf{H}(\mathbf{u})\} \\ \mathbf{H}(\mathbf{u}) &= \nabla_{\mathbf{u}\mathbf{u}}g(\mathbf{u})\end{aligned}$$



# 1 forward $\rightarrow$ 2 reverse sweeps

- Find  $\nabla_{\mathbf{u}}\psi(u)$ , where

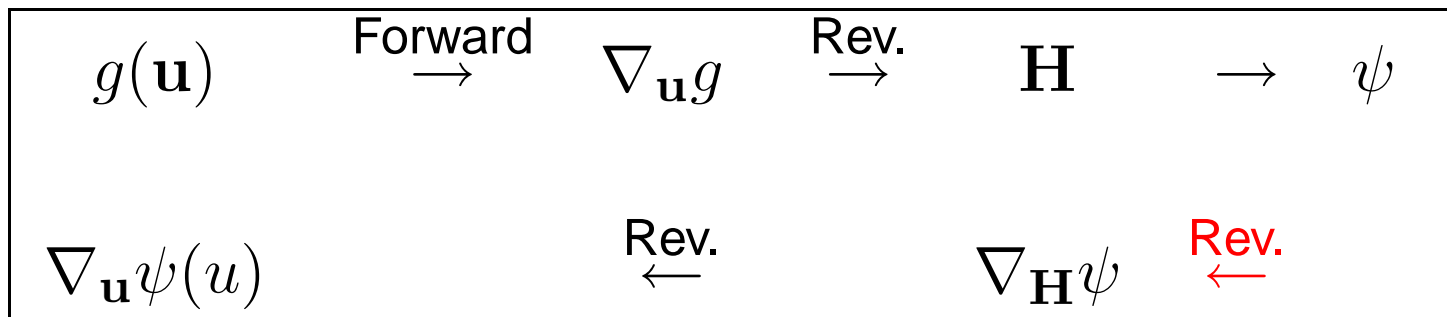
$$\begin{aligned}\psi(\mathbf{u}) &= \det \{\mathbf{H}(\mathbf{u})\} \\ \mathbf{H}(\mathbf{u}) &= \nabla_{\mathbf{u}\mathbf{u}}g(\mathbf{u})\end{aligned}$$



# 1 forward $\rightarrow$ 2 reverse sweeps

- Find  $\nabla_{\mathbf{u}}\psi(u)$ , where

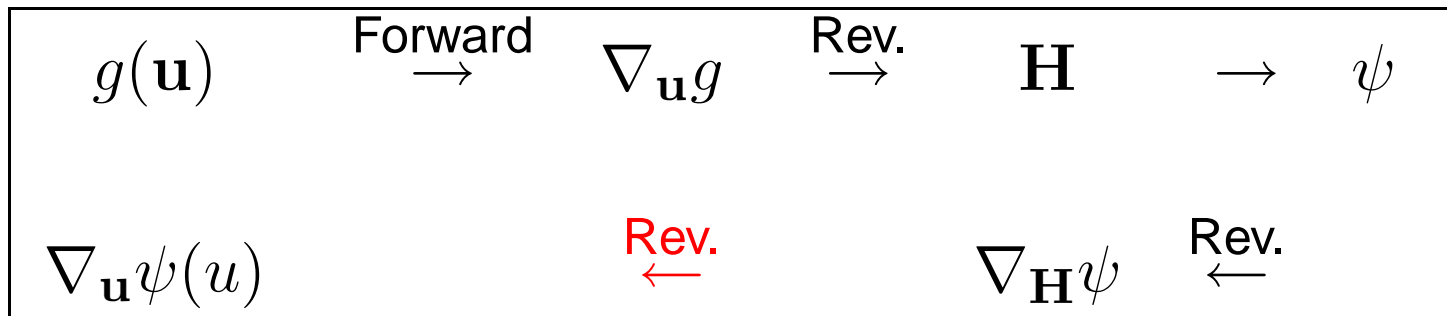
$$\begin{aligned}\psi(\mathbf{u}) &= \det \{\mathbf{H}(\mathbf{u})\} \\ \mathbf{H}(\mathbf{u}) &= \nabla_{\mathbf{u}\mathbf{u}}g(\mathbf{u})\end{aligned}$$



# 1 forward $\rightarrow$ 2 reverse sweeps

- Find  $\nabla_{\mathbf{u}}\psi(u)$ , where

$$\begin{aligned}\psi(\mathbf{u}) &= \det \{\mathbf{H}(\mathbf{u})\} \\ \mathbf{H}(\mathbf{u}) &= \nabla_{\mathbf{u}\mathbf{u}}g(\mathbf{u})\end{aligned}$$



# *Implementation in AD Model Builder*

- AD Model Builder is David Fournier's system for parameter estimation
  - Model formulation in C++
  - Reverse mode AD + quasi-Newton
  - Operator overloading in C++

# *Partial separability*

- Means in practice that  $\mathbb{H}$  is sparse (banded, block diagonal, etc)

# *Partial separability*

- Means in practice that  $\mathbb{H}$  is sparse (banded, block diagonal, etc)
- Known as "conditional independence" in statistics

# *Partial separability*

- Means in practice that  $\mathbb{H}$  is sparse (banded, block diagonal, etc)
- Known as "conditional independence" in statistics
- We have seen an example: Polio data set

## *Final remarks*

- AD useful in statistics
- AD + Laplace approximation  $\Rightarrow$  automatic evaluation of likelihood
- Statistics today dominated by Monte Carlo methods
  - Difficult to judge convergence
  - Exact gradient  $\nabla_{\theta} L^*(\theta)$  is needed
- More information about the software:

<http://otter-rsch.com/admodel.htm>