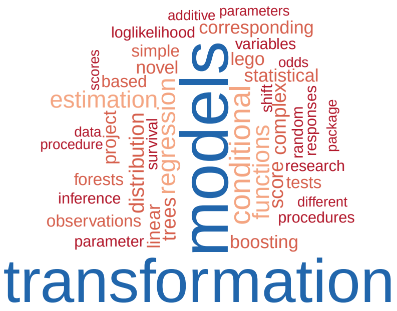
University of Zurich^UZH

EBPI Epidemiology, Biostatistics and Prevention Institute

useR! 2019 Tutorial: Transformation Models

Torsten Hothorn



Toulouse, 2019-07-09

Introduction

The Central Dogma of Statistics

Everything is in the distribution:

$$Y \sim \mathbb{P}_Y$$

The random variable Y is called response (outcome, dependent, endogenous) variable.

Q: How can we obtain an estimate $\hat{\mathbb{P}}_Y$ from observations $Y_1, \dots, Y_N$?

Regression Analysis

Everything is in the *conditional* distribution:

$$Y \mid \boldsymbol{X} = \boldsymbol{x} \sim \mathbb{P}_{Y \mid \boldsymbol{X} = \boldsymbol{x}}$$

$\boldsymbol{X}$ (typically multivariate) are called explanatory (design, independent, exogenous, predictor) variables or covariates.

Q: How do changes in $\boldsymbol{x}$ propagate to changes in $\mathbb{P}_{Y \mid \boldsymbol{X} = \boldsymbol{x}}$?

Q: How can we estimate $\hat{\mathbb{P}}_{Y \mid \boldsymbol{X} = \boldsymbol{x}}$ from $(Y_1, \boldsymbol{x}_1), \dots, (Y_N, \boldsymbol{x}_N)$?

Objectives for Today

- Learn about / recap transformation models for $\mathbb{P}_Y$ and $\mathbb{P}_{Y|\boldsymbol{X}=\boldsymbol{x}}$
- Interpret parameters of important models in this class
- Use maximum-likelihood (ML) estimation to estimate parameters
- Sample from models
- Assess model quality
- Improve model quality

R Add-on Packages (CRAN)

- **mlt** ("most likely transformations"): the workhorse for ML estimation (uses **basefun** and **variables**)
- **mlt.docreg**: vignette and tests
- **tram** ("transformation models"): formula-based user interfaces to specific transformation models, with vignette torturing the Boston Housing data
- **tbm**: transformation boosting machines
- **trtf**: transformation trees and forests

Resources

- http://ctm.R-forge.R-project.org
- "Most Likely Transformations", SJoS, **mlt**, **tram**, http://doi.org/10.1111/sjos.12291
- "Conditional Transformation Models", JRSS-B, **tbm**, http://doi.org/10.1111/rssb.12017
- "(Survival) Transformation Forests", **trtf**, https://arxiv.org/abs/1701.02110, https://arxiv.org/abs/1902.01587
- "Top-Down Transformation Choice", SM, **trtf**, http://arxiv.org/abs/1706.08269
- "Transformation Boosting Machines", STCO, **tbm**, http://doi.org/10.1007/s11222-019-09870-4

Trigger Warning

- The material will question some things most stats people take for granted.
- "Complex" models will look rather simple.
- Terms will be used in more generic ways than usual.
- Code is slow.
- Code is memory inefficient.
-

Trigger Warning

- The material will question some things most stats people take for granted.
- “Complex” models will look rather simple.
- Terms will be used in more generic ways than usual.
- Code is slow.
- Code is memory inefficient.
- Get a new computer if this is a problem.

Illustration: Body Mass Index (BMI) Distributions

2012 survey ($N = 16427$) in Switzerland
Explain conditional distribution of BMI given $\boldsymbol{x} =$

- Sex,
- Smoking status,
- Age,
- Education,
- Physical activity,
- Alcohol intake,
- Fruit and vegetable consumption,
- Region, and
- Nationality.

Illustration: Body Mass Index (BMI)

Recall that BMI is defined as

$$Y \quad := \quad \frac{\text{weight (in kg)}}{(\text{height (in cm)})^2}$$

Maybe use a Normal Linear Regression Model (NLRM)

$$Y \quad = \quad \tilde{\alpha} + \boldsymbol{x}^\top \tilde{\boldsymbol{\beta}} + \sigma\varepsilon, \quad \varepsilon \sim \text{N}(0, 1)$$

WHO Categories for BMI

The WHO defines the BMI categories underweight ($\text{BMI}_{18.5} = I(\text{BMI} \le 18.5)$), normal weight ($\text{BMI}_{(18.5,25]} = I(18.5 < \text{BMI} \le 25)$), overweight ($\text{BMI}_{(25,30]} = I(25 < \text{BMI} \le 30)$), and obese ($\text{BMI} > 30$).

Maybe use Proportional Odds Logistic Regression (POLR):

$$\text{logit}(\mathbb{P}(Y \le y_k \mid \boldsymbol{x})) = \vartheta_k + \boldsymbol{x}^\top \boldsymbol{\beta}, k = 1, \dots, 3$$

with $y_1 = 18.5, y_2 = 25, y_3 = 30$

Illustration: Disease-free Survival Time

CAO/ARO/AIO-04 trial, explain disease-free survival time $T > 0$ of rectal cancer patients given two treatments (and other baseline variables) $\boldsymbol{x}$

Maybe use Cox' Proportional Hazards Model (Cox):

$$\lambda(t \mid \boldsymbol{x}) = \lambda_0(t) \exp(\boldsymbol{x}^\top \beta)$$

Classical Compartments

It seems we need three books (lectures, tutorials, ...)

- "Regression" Analysis
- Survival Analysis
- Analysis of Ordered Categorical Data

Transformation models hit all birds with one stone.

Rearranging POLR

$$\begin{aligned}
\operatorname{logit}(\mathbb{P}(Y \le y_k \mid \boldsymbol{x})) &= \vartheta_k + \boldsymbol{x}^\top \beta \\
\operatorname{logit}(\mathbb{P}(Y \le y_k \mid \boldsymbol{x})) &= (0, \dots, 1, \dots, 0)(\vartheta_1, \dots, \vartheta_{K-1})^\top + \boldsymbol{x}^\top \beta \\
\operatorname{logit}(\mathbb{P}(Y \le y_k \mid \boldsymbol{x})) &= \boldsymbol{a}_{\text{POLR}}(y_k)^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \beta \\
\mathbb{P}(Y \le y_k \mid \boldsymbol{x}) &= \operatorname{logit}^{-1}(\boldsymbol{a}_{\text{POLR}}(y_k)^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \beta)
\end{aligned}$$

Constraint: $\boldsymbol{a}_{\text{POLR}}(y_k)^\top \boldsymbol{\vartheta} \le \boldsymbol{a}_{\text{POLR}}(y_{k+1})^\top \boldsymbol{\vartheta}$

Rearranging Cox

$$\begin{aligned}
\lambda(t \mid \boldsymbol{x}) &= \lambda_0(t) \exp(\boldsymbol{x}^\top \beta) \\
\Lambda(t \mid \boldsymbol{x}) &= \Lambda_0(t) \exp(\boldsymbol{x}^\top \beta), \quad \Lambda_0(t) = \int_0^t \lambda_0(u) du \\
1 - \mathbb{P}(T \le t \mid \boldsymbol{x}) &= \exp(-\Lambda_0(t) \exp(\boldsymbol{x}^\top \beta)) \\
\mathbb{P}(T \le t \mid \boldsymbol{x}) &= 1 - \exp(-\exp(\log(\Lambda_0(t)) + \boldsymbol{x}^\top \beta))) \\
\mathbb{P}(T \le t \mid \boldsymbol{x}) &= \operatorname{cloglog}^{-1}(\log(\Lambda_0(t)) + \boldsymbol{x}^\top \beta)) \\
\mathbb{P}(Y \le y \mid \boldsymbol{x}) &= \operatorname{cloglog}^{-1}(\boldsymbol{a}_{\text{Cox}}(y)^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \beta)
\end{aligned}$$

Constraint: $\boldsymbol{a}_{\text{Cox}}(y)^\top \boldsymbol{\vartheta} \le \boldsymbol{a}_{\text{Cox}}(y + \epsilon)^\top \boldsymbol{\vartheta}$ for all $\epsilon > 0$

Rearranging NLRM

$$\begin{aligned}
Y &= \tilde{\alpha} + \boldsymbol{x}^\top \tilde{\boldsymbol{\beta}} + \sigma\varepsilon, \quad \varepsilon \sim \text{N}(0,1) \\
\sigma^{-1} Y &= \sigma^{-1}\tilde{\alpha} + \boldsymbol{x}^\top \sigma^{-1}\tilde{\boldsymbol{\beta}} + \varepsilon \\
\sigma^{-1} Y &= \alpha + \boldsymbol{x}^\top \boldsymbol{\beta} + \varepsilon \\
\mathbb{P}(Y \le y \mid \boldsymbol{x}) &= \Phi(\sigma^{-1} y - \alpha - \boldsymbol{x}^\top \boldsymbol{\beta}) \\
\mathbb{P}(Y \le y \mid \boldsymbol{x}) &= \Phi((y, -1)(\sigma^{-1}, \alpha)^\top - \boldsymbol{x}^\top \boldsymbol{\beta}) \\
\mathbb{P}(Y \le y \mid \boldsymbol{x}) &= \text{probit}^{-1}(\boldsymbol{a}_{\text{NLRM}}(y)^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta})
\end{aligned}$$

Constraint: $\sigma > 0$

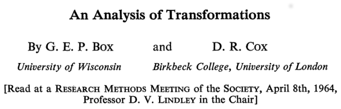
Linear Transformation Model

$$\mathbb{P}(Y \le y \mid \boldsymbol{x}) \quad = \quad \text{link}^{-1}(h(y) \pm \boldsymbol{x}^\top \boldsymbol{\beta}) = \text{link}^{-1}(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} \pm \boldsymbol{x}^\top \boldsymbol{\beta})$$

Constraint: $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$ monotone non-decreasing

Note: The transformation $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$ is potentially non-linear, the name refers to the linear predictor $\boldsymbol{x}^\top \boldsymbol{\beta}$

An Analysis of Transformations (1964)



An Analysis of Transformations

By G. E. P. Box and D. R. Cox

University of Wisconsin *Birkbeck College, University of London*

[Read at a Research Methods Meeting of the Society, April 8th, 1964,
Professor D. V. Lindley in the Chair]

$$h(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \lambda > 0 \\ \log(y) & \lambda = 0 \end{cases}$$

for $F_Z = \Phi$ and $Y \in \mathbb{R}$.

Link Functions

Link functions are quantile functions F_Z^{-1} of a latent variable $Z \in \mathbb{R}$

Link	Name	$F_Z(z)$	$F_Z^{-1}(p)$
probit	Normal	$\Phi(z)$	$\Phi^{-1}(p)$
logit	Logistic	$(1 + \exp(-z))^{-1}$	$\log(p/(1-p))$
cloglog	Gompertz	$1 - \exp(-\exp(z))$	$\log(-\log(1-p))$
loglog	Gumbel	$\exp(-\exp(-z))$	$-\log(\log(p))$

Densities $f_Z(z) = F_Z'(z)$ are log-concave

$$\mathbb{P}(Y \le y \mid \boldsymbol{x}) \quad = \quad F_{Y \mid \boldsymbol{X} = \boldsymbol{x}}(y \mid \boldsymbol{x}) = F_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \boldsymbol{\beta})$$

Probabilities and Densities

Probabilities

$$\mathbb{P}(\underline{y} < Y \leq \bar{y} \mid \boldsymbol{x}) = F_Z(\boldsymbol{a}(\bar{y})^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \boldsymbol{\beta}) - F_Z(\boldsymbol{a}(\underline{y})^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \boldsymbol{\beta})$$

Discrete densities

$$f_{Y \mid \boldsymbol{X} = \boldsymbol{x}}(y_k \mid \boldsymbol{x}) = F_Z(\boldsymbol{a}(y_k)^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \boldsymbol{\beta}) - F_Z(\boldsymbol{a}(y_{k-1})^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \boldsymbol{\beta})$$

Absolute continuous densities

$$f_{Y \mid \boldsymbol{X} = \boldsymbol{x}}(y_k \mid \boldsymbol{x}) = F'_{Y \mid \boldsymbol{X} = \boldsymbol{x}}(y \mid \boldsymbol{x}) = f_Z(\boldsymbol{a}(y_k)^\top \boldsymbol{\vartheta} + \boldsymbol{x}^\top \boldsymbol{\beta}) \boldsymbol{a}'(y)^\top \boldsymbol{\vartheta}$$

Model Interpretation

Let $\boldsymbol{x}_0$ such that $\boldsymbol{x}_0^\top \boldsymbol{\beta} = 0$
probit:

$$\mathbb{E}(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} \mid \boldsymbol{x}) - \mathbb{E}(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} \mid \boldsymbol{x}_0) = \boldsymbol{x}^\top \boldsymbol{\beta}$$

logit:

$$\frac{\mathbb{P}(Y \leq y \mid \boldsymbol{x})}{\mathbb{P}(Y > y \mid \boldsymbol{x})} = \exp(\boldsymbol{x}^\top \boldsymbol{\beta}) \frac{\mathbb{P}(Y \leq y \mid \boldsymbol{x}_0)}{\mathbb{P}(Y > y \mid \boldsymbol{x}_0)} = \exp(\boldsymbol{x}^\top \boldsymbol{\beta}) \exp(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta})$$

cloglog:

$$\mathbb{P}(Y > y \mid \boldsymbol{x}) = \mathbb{P}(Y > y \mid \boldsymbol{x}_0)^{\exp(\boldsymbol{x}^\top \boldsymbol{\beta})} = \exp(-\exp(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta}))^{\exp(\boldsymbol{x}^\top \boldsymbol{\beta})}$$

loglog:

$$\mathbb{P}(Y \leq y \mid \boldsymbol{x}) = \mathbb{P}(Y \leq y \mid \boldsymbol{x}_0)^{\exp(\boldsymbol{x}^\top \boldsymbol{\beta})}$$

Model Definition

1. Pick F_Z to define scale of $\boldsymbol{\beta}$
2. Define transformation of response
   $h(Y) = \boldsymbol{a}(Y)^\top \boldsymbol{\vartheta} =: Z \sim F_Z$ such that

$$\begin{aligned} \mathbb{P}(Y \leq y \mid \boldsymbol{x}_0) &= F_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta}) \\ F_Z^{-1}(\mathbb{P}(Y \leq y \mid \boldsymbol{x}_0)) &= h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta} \end{aligned}$$

   by choosing suitable basis functions $\boldsymbol{a} : \Xi \rightarrow \mathbb{R}^P$ for a response $Y \in \Xi$ (pay attention to bounds etc. here)
3. Define sign of $\boldsymbol{x}^\top \boldsymbol{\beta}$

Note: Flexible enough basis functions $\boldsymbol{a}$ can generate *all* distribution functions $\mathbb{P}(Y \leq y \mid \boldsymbol{x}_0)$!

Model Estimation

- Maximum likelihood estimation to obtain $\hat{\boldsymbol{\vartheta}}$ and $\hat{\boldsymbol{\beta}}$ simultaneously
- Thus, the most likely transformation $\boldsymbol{a}(y)^\top \hat{\boldsymbol{\vartheta}}$ *and* the regression coefficients $\hat{\boldsymbol{\beta}}$ are estimated jointly. No need to transform observations manually before estimating a model
- Suitable constraints on $\boldsymbol{\vartheta}$ to ensure a monotone non-decreasing $\boldsymbol{a}(y)^\top \hat{\boldsymbol{\vartheta}}$
- Hessian for $\boldsymbol{\vartheta}$ and $\boldsymbol{\beta}$ for asymptotic normal inference
- Score-based inference for exact conditional inference and statistical learning

The tram Package

Formula-based user interface to some specific transformation models, including

- `Lm()`: A beefed-up version of `lm()` assuming a conditional normal response Y
- `BoxCox()`: Extension of `Lm()` for non-normal Y
- `Coxph()`: Fully parametric Cox models
- `Polr()`: A beefed-up version of `MASS:polr()`
- `Colr()`: Continuous outcome logistic regression
- `Lehmann()`: Regression for Lehmann alternatives

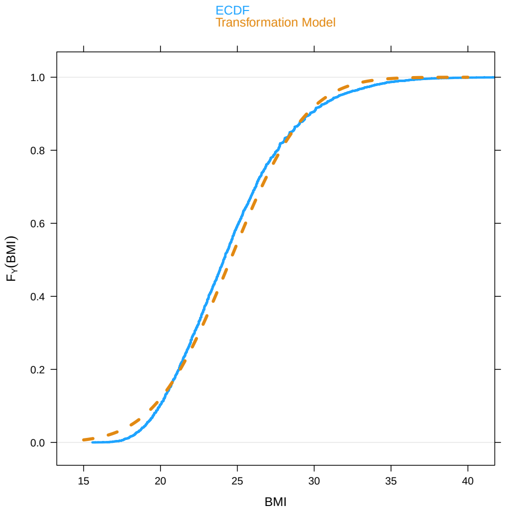
```
library("tram")
```

Unconditional Normal Model for BMI Distribution

$$\mathbb{P}(Y \le y) = \Phi((1, y)^\top \vartheta) \iff Y \sim \mathrm{N}(-\vartheta_1 \vartheta_2^{-1}, \vartheta_2^{-2})$$

```
logLik(mLm <- Lm(bmi ~ 1, data = SGB12, weights = wght))

## 'log Lik.' -14229721 (df=2)

(cf <- coef(as.mlt(mLm)))

## (Intercept)         bmi
##  -6.3253920   0.2576529

-cf[1] / cf[2]

## (Intercept)
##    24.55005

weighted.mean(SGB12$bmi, SGB12$wght)

## [1] 24.55005
```

Unconditional Normal Model for BMI Distribution



Unconditional Non-Normal Model for BMI Distribution

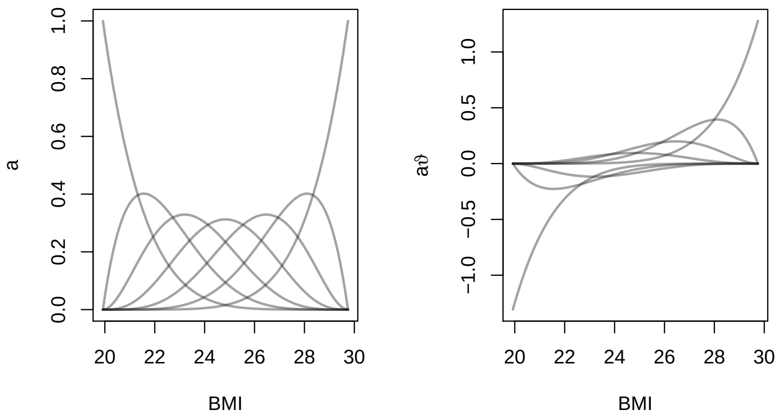
$$\mathbb{P}(Y \le y) = \Phi(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \vartheta)$$

```
logLik(mBC <- BoxCox(bmi ~ 1, data = SGB12, weights = wght))

## 'log Lik.' -13987606 (df=7)

coef(as.mlt(mBC))

##    Bs1(bmi)    Bs2(bmi)    Bs3(bmi)    Bs4(bmi)    Bs5(bmi)    Bs6(bmi)
## -1.3076861 -0.5644059 -0.3563150  0.3046087  0.6040538  0.9820012
##    Bs7(bmi)
##  1.2778463
```

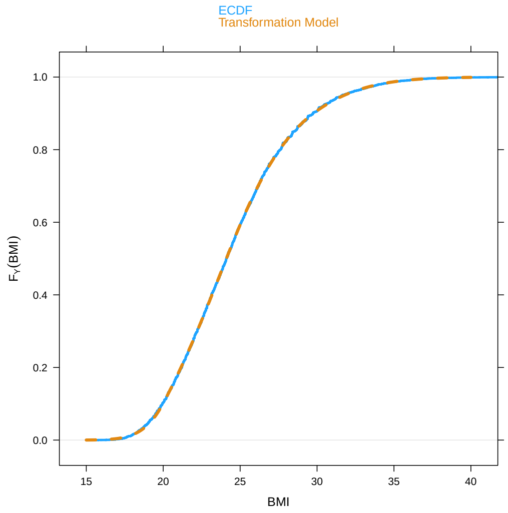
The name of this function was chosen to honor the first paper on transformation models (Box and Cox, 1964, JRSS-B, who suggested a power transformation $h(Y) \sim \mathrm{N}(\alpha + \boldsymbol{x}^\top \vartheta, \sigma^2)$). `BoxCox()` *DOES NOT* apply this power transformation but a Bernstein polynomial.

Basis Functions for BMI

$P = 7$ Bernstein basis functions $\boldsymbol{a}_{\text{Bs},P-1}(y)$ on some interval (with linear extrapolation outside)



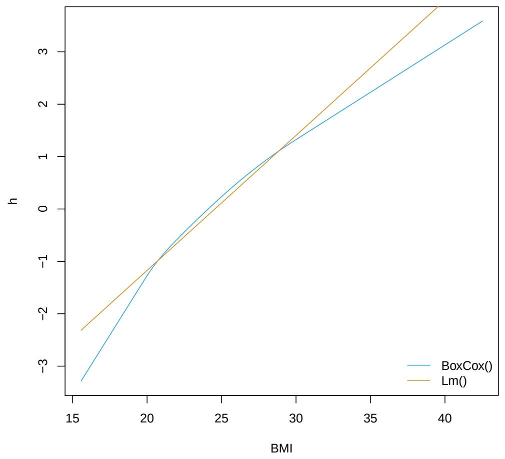
Unconditional Non-Normal Model for BMI Distribution



Only $P = 7$ parameters needed to recover the ECDF (16427 parameters)

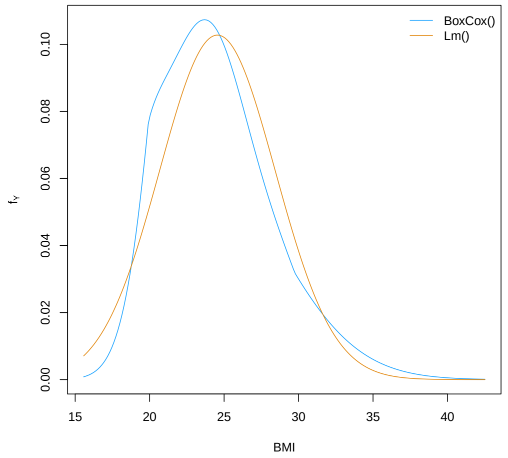
Linear vs. Non-linear Transformation of BMI

$$h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$$



Linear vs. Non-linear Transformation of BMI

$$f_Y(y) = \phi(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta}) \boldsymbol{a}'_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta}$$



Link Functions Don’t Matter Unconditionally

```
logLik(mLm)

## 'log Lik.' -14229721 (df=2)

logLik(BoxCox(bmi ~ 1, data = SGB12, weights = wght, order = 25))

## 'log Lik.' -13985142 (df=26)

logLik(Coxph(bmi ~ 1, data = SGB12, weights = wght, order = 25))

## 'log Lik.' -14004147 (df=26)

logLik(Colr(bmi ~ 1, data = SGB12, weights = wght, order = 25))

## 'log Lik.' -14000323 (df=26)

logLik(Lehmann(bmi ~ 1, data = SGB12, weights = wght, order = 25))

## 'log Lik.' -13982410 (df=26)
```

BMI Reconsidered

Recall

$$Y := \frac{\text{weight (in kg)}}{(\text{height (in cm)})^2}$$

For an individual 1.75m tall weighting 76kg, all BMI values between $75.5/1.755^2 = 24.51$ and $76.5/1.745^2 = 25.12$ are possible due to rounding error.
The “interval-censored” likelihood contribution is

$$\mathbb{P}(24.51 < Y \le 25.12) = \Phi(\boldsymbol{a}_{\text{Bs},P-1}(25.13)^\top \vartheta) - \Phi(\boldsymbol{a}_{\text{Bs},P-1}(24.51)^\top \vartheta)$$

`Surv()` objects representing censored (or truncated) observations can be specified as LHS of a formula.

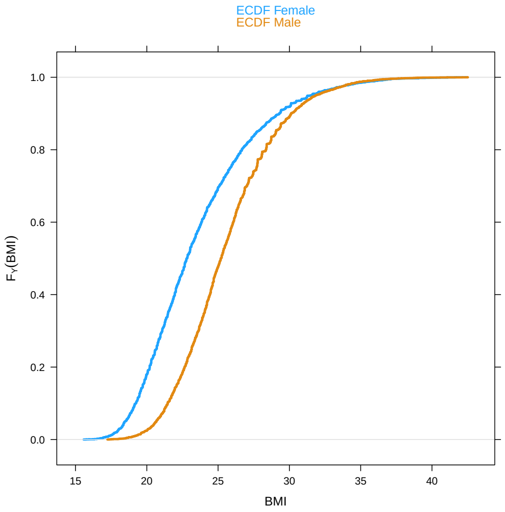
BMI Reconsidered

The WHO defines the BMI categories underweight ($\text{BMI}_{18.5} = I(\text{BMI} \le 18.5)$), normal weight ($\text{BMI}_{(18.5,25]} = I(18.5 < \text{BMI} \le 25)$), overweight ($\text{BMI}_{(25,30]} = I(25 < \text{BMI} \le 30)$), and obese ($\text{BMI} > 30$).

```
cumsum(prop.table(xtabs(wght ~ bmiWHO, data = SGB12)))

##   [0,18.5)  [18.5,25)    [25,30)   [30,100)
## 0.02790122 0.59225836 0.90650778 1.00000000

mP <- Polr(bmiWHO ~ 1, data = SGB12, weights = wght)
plogis(coef(as.mlt(mP)))

##    bmiWHO1    bmiWHO2    bmiWHO3
## 0.02790122 0.59225836 0.90650778

predict(mBC, newdata = data.frame(bmi = c(18.5, 25, 30)),
        type = "distribution")

##          1          2          3
## 0.02540007 0.59281796 0.90721119
```

Two-group Comparisons

BMI Distribution in Females and Males



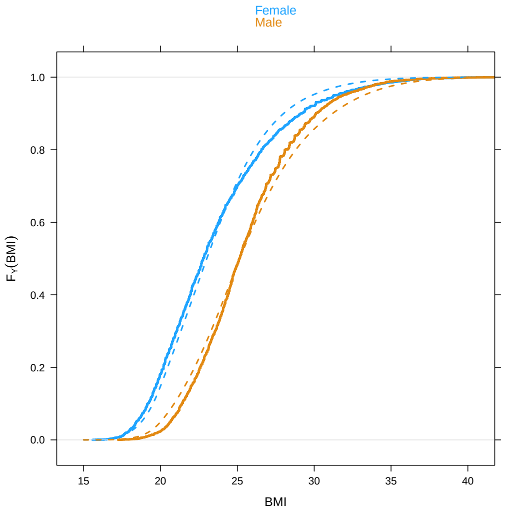
Shift After Non-linear Transformation

$$\mathbb{P}(Y \le y \mid \text{sex}) = \Phi(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta} - \beta \mathbb{1}(\text{male}))$$

```
logLik(m1BC <- BoxCox(bmi ~ sex, data = SGB12, weights = wght))

## 'log Lik.' -13767163 (df=8)

coef(m1BC)

##   sexMale
## 0.6028754

confint(m1BC)

##              2.5 %    97.5 %
## sexMale 0.6010948 0.6046561
```

Interpretation: $\mathbb{E}(h(Y) \mid \text{male}) - \mathbb{E}(h(Y) \mid \text{female}) = 0.603$
Hm.

Shift After Non-linear Transformation



Shift After Linear Transformation

$$\mathbb{P}(Y \le y \mid \text{sex}) = \Phi(\xi y - \alpha - \beta \mathbb{1}(\text{male}))$$

```
logLik(m1Lm <- Lm(bmi ~ sex, data = SGB12, weights = wght))

## 'log Lik.' -14060496 (df=3)

coef(m1Lm)

##  sexMale
## 0.522452

confint(m1Lm)

##              2.5 %    97.5 %
## sexMale 0.5206916 0.5242125
```

Interpretation:
$\mathbb{E}(\xi Y - \alpha \mid \text{male}) - \mathbb{E}(\xi Y - \alpha \mid \text{female}) = 0.522$
Hm.

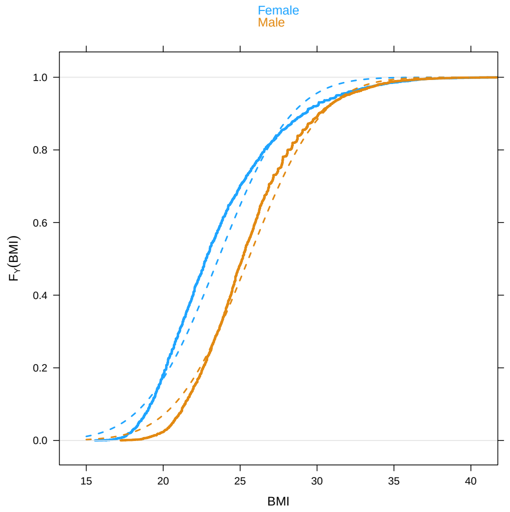
Shift After Linear Transformation

$$\mathbb{P}(Y \le y \mid \text{sex}) = \Phi(\xi y - \alpha - \beta \mathbb{1}(\text{male}))$$

```
(mcf <- coef(m1Lm) / coef(as.mlt(m1Lm))["bmi"])

## sexMale
## 1.96191

coef(lm(bmi ~ sex, data = SGB12, weights = wght))

## (Intercept)     sexMale
##   23.579607    1.961909
```

Interpretation: $\mathbb{E}(Y \mid \text{male}) - \mathbb{E}(Y \mid \text{female}) = \xi^{-1}\beta = 1.962$

Shift After Linear Transformation



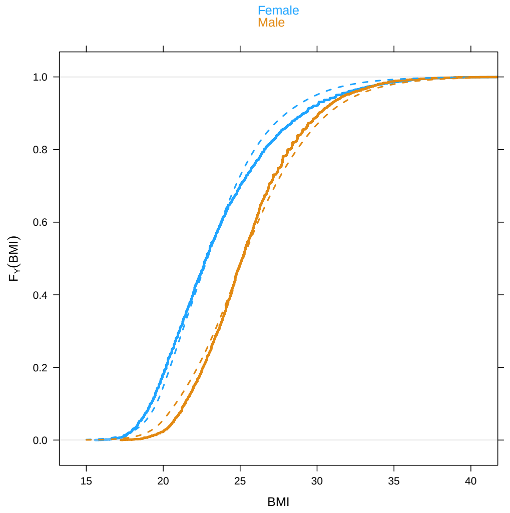
Odds Ratio Alternatives

$$\mathbb{P}(Y \le y \mid \text{sex}) = \text{logit}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta} + \beta \mathbb{1}(\text{male}))$$

```
logLik(m1C <- Colr(bmi ~ sex, data = SGB12, weights = wght))

## 'log Lik.' -13773675 (df=8)

c(coef(m1C), confint(m1C))

##   sexMale
## -1.069573 -1.072698 -1.066449

exp(c(coef(m1C), confint(m1C)))

##   sexMale
## 0.3431550 0.3420845 0.3442289
```

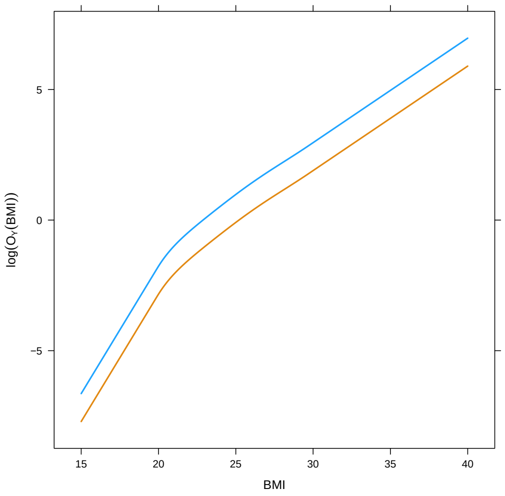
Interpretation $\frac{\mathbb{P}(Y \le y | \text{male})}{\mathbb{P}(Y > y | \text{male})} = \exp(\beta) \frac{\mathbb{P}(Y \le y | \text{female})}{\mathbb{P}(Y > y | \text{female})}$

Odds Ratio Alternatives



Odds Ratio Alternative

$h(y) + \beta \mathbb{1}(\text{male})$ is conditional log-odds function



Odds Ratio Alternative

The model can be understood as a series of logistic binary regression models

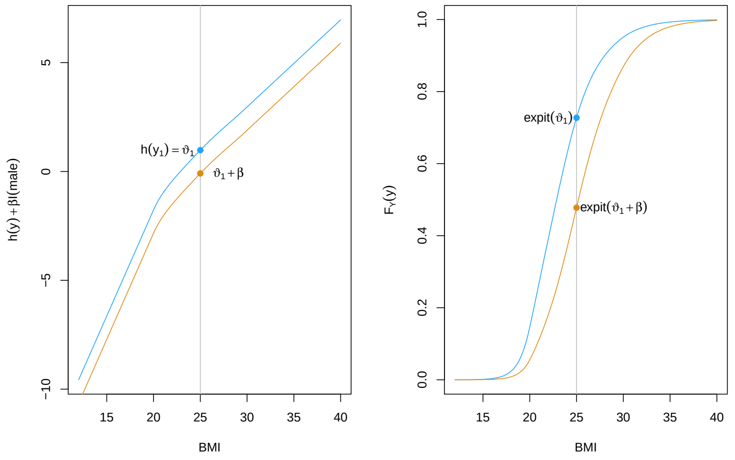
$$\text{logit}(\mathbb{E}(\mathbb{1}(Y \le y) \mid \text{sex})) = \alpha(y) + \beta \mathbb{1}(\text{male})$$

where *ONLY* the intercept α is allowed to vary with the cut-off point y used to dichotomise the response values

In this light, `BoxCox()` is a series of probit models; `Coxph()` and `Lehmann()` use cloglog and loglog link funcions.

Colr() vs. glm()

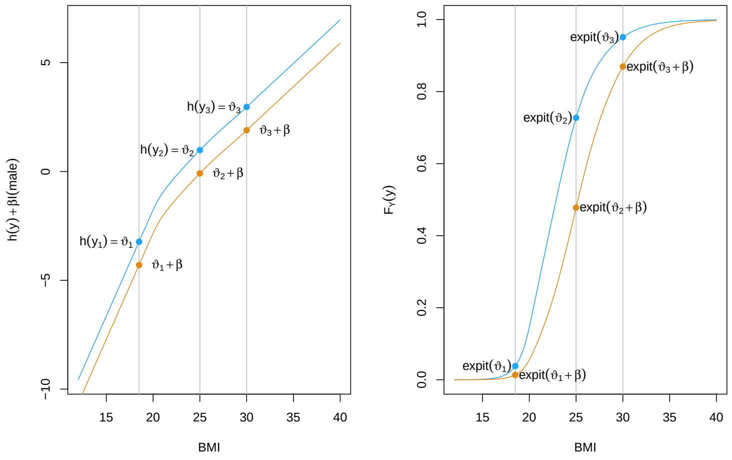
$\text{logit}(\mathbb{P}(\text{BMI} \le 25 \mid \text{sex})) = \vartheta_1 + \beta \mathbb{1}(\text{male}))$



BMI $\le$ 25 means underweight or normal

Colr() vs. polr()

$\mathbb{P}(\text{BMI} \le y_k \mid \text{sex}) = \text{expit}(\vartheta_k + \beta \mathbb{1}(\text{male}))$



BMI $\le$ 18.5 means underweight; BMI $\le$ 30 means not obese

Colr() vs. polr() vs. glm()

```
exp(c(coef(m1C), confint(m1C)))

##   sexMale
## 0.3431550 0.3420845 0.3442289

library("MASS")
m1P <- polr(bmiWHO ~ sex, data = SGB12, weights = wght)
exp(-c(coef(m1P), confint(m1P)))

##   sexMale     2.5 %    97.5 %
## 0.4020473 0.4034335 0.4006652

m1L <- glm(I(bmi < 25) ~ sex, data = SGB12, weights = wght, family = binomial())
exp(c(coef(m1L)["sexMale"], confint(m1L)["sexMale",]))

##   sexMale     2.5 %    97.5 %
## 0.3993755 0.3979312 0.4008248
```

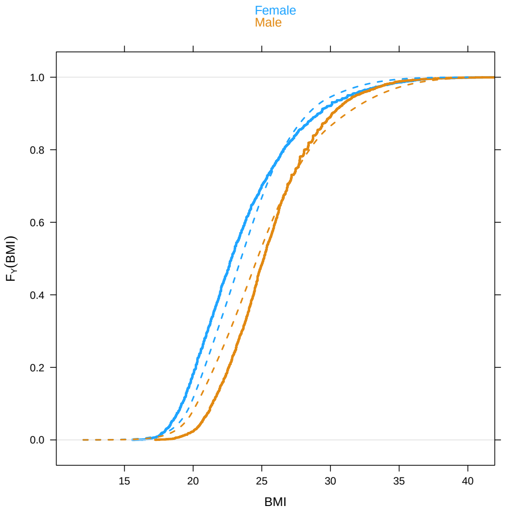
Hazard Ratio Alternatives

$$\mathbb{P}(Y \le y \mid \text{sex}) = \text{cloglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta} + \beta \mathbb{1}(\text{male}))$$

```
logLik(m1Cx <- Coxph(bmi ~ sex, data = SGB12, weights = wght))

## 'log Lik.' -13923093 (df=8)

exp(c(coef(m1Cx), confint(m1Cx)))

##   sexMale
## 0.6895182 0.6883216 0.6907169
```

Interpretation: $\mathbb{P}(Y > y \mid \text{male}) = \mathbb{P}(Y > y \mid \text{female})^{\exp(\beta)}$

Hazard Ratio Alternatives



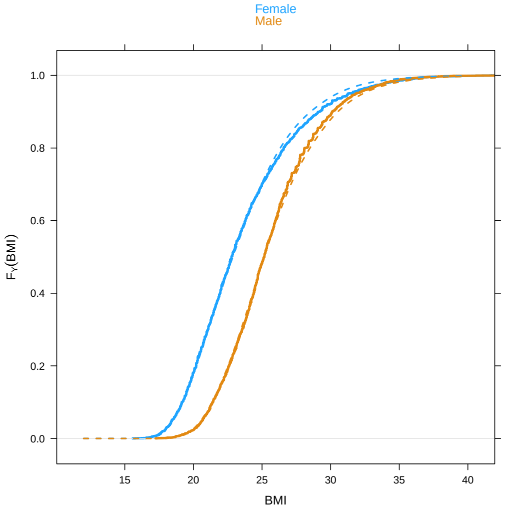
Lehmann Alternatives

$$\mathbb{P}(Y \le y \mid \text{sex}) = \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta} - \beta \mathbb{1}(\text{male}))$$

```
logLik(m1L <- Lehmann(bmi ~ sex, data = SGB12, weights = wght))

## 'log Lik.' -13677664 (df=8)

exp(-c(coef(m1L), confint(m1L)))

##   sexMale
## 0.4794473 0.4803268 0.4785695
```

Interpretation: $\mathbb{P}(Y \le y \mid \text{male}) = \mathbb{P}(Y \le y \mid \text{female})^{\exp(-\beta)}$

Lehmann Alternatives

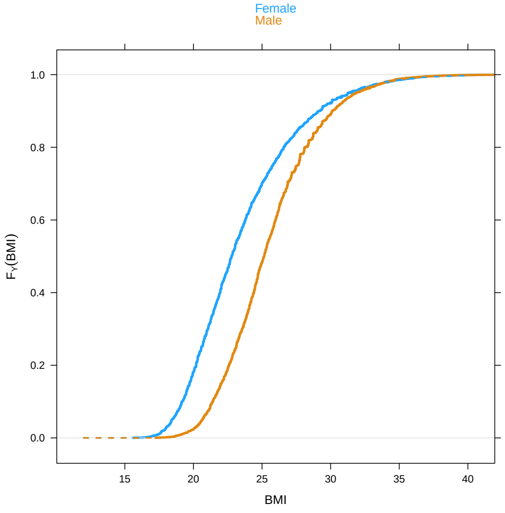


Can We Do Better?

$$\mathbb{P}(Y \leq y \mid \text{sex}) = \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta}(\text{sex}))$$

```
logLik(m1Ls <- Lehmann(bmi | 0 + sex ~ 1, data = SGB12, weights = wght))

## 'log Lik.' -13656590 (df=14)

coef(as.mlt(m1Ls))

## Bs1(bmi):sexFemale Bs2(bmi):sexFemale Bs3(bmi):sexFemale
##        -0.56753762         0.04700422         0.39081410
## Bs4(bmi):sexFemale Bs5(bmi):sexFemale Bs6(bmi):sexFemale
##         1.12128126         1.40184337         1.89864895
## Bs7(bmi):sexFemale   Bs1(bmi):sexMale   Bs2(bmi):sexMale
##         2.46252460        -1.31516543        -0.83168833
##   Bs3(bmi):sexMale   Bs4(bmi):sexMale   Bs5(bmi):sexMale
##        -0.13912522        -0.13912522         1.13200712
##   Bs6(bmi):sexMale   Bs7(bmi):sexMale
##         1.37929723         2.07889045
```

Can We Do Better?



Stratification

$$\mathbb{P}(Y \leq y \mid \text{sex}) = \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y)^\top \boldsymbol{\vartheta}(\text{sex}))$$

means we are estimating two transformation functions, one for males and one for females.

This is called *stratification* in classical terms.

Parameterisation:
$\mathbb{P}(Y \leq y \mid \text{sex}) = \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y) \otimes \boldsymbol{b}(\text{sex})^\top \boldsymbol{\vartheta}))$ with

$$\boldsymbol{b}(\text{sex}) = \begin{cases} (1,0) & : \text{female} \\ (0,1) & : \text{male} \end{cases} \text{or} \begin{cases} (1,0) & : \text{female} \\ (1,1) & : \text{male} \end{cases}$$

Stratification

```
logLik(m1Ls)

## 'log Lik.' -13656590 (df=14)

logLik(m2Ls <- Lehmann(bmi | sex ~ 1, data = SGB12, weights = wght))

## 'log Lik.' -13656590 (df=14)

cbind(coef(as.mlt(m2Ls)), confint(as.mlt(m2Ls)))

##                                         2.5 %      97.5 %
## Bs1(bmi):(Intercept) -0.56751356 -0.56891337 -0.56611375
## Bs2(bmi):(Intercept)  0.04701838  0.04482116  0.04921559
## Bs3(bmi):(Intercept)  0.39074307  0.38344306  0.39804308
## Bs4(bmi):(Intercept)  1.12142160  1.10953008  1.13331312
## Bs5(bmi):(Intercept)  1.40175638  1.39184549  1.41166728
## Bs6(bmi):(Intercept)  1.89877569  1.89435380  1.90319758
## Bs7(bmi):(Intercept)  2.46258029  2.45854952  2.46661105
## Bs1(bmi):sexMale     -0.74763508 -0.74991179 -0.74535837
## Bs2(bmi):sexMale     -0.87871694 -0.88256875 -0.87486513
## Bs3(bmi):sexMale     -0.52985497 -0.54123740 -0.51847255
## Bs4(bmi):sexMale     -1.26053337 -1.27780087 -1.24326586
## Bs5(bmi):sexMale     -0.26971858 -0.28339764 -0.25603952
## Bs6(bmi):sexMale     -0.51936250 -0.52523494 -0.51349006
## Bs7(bmi):sexMale     -0.38355355 -0.38880404 -0.37830306
```

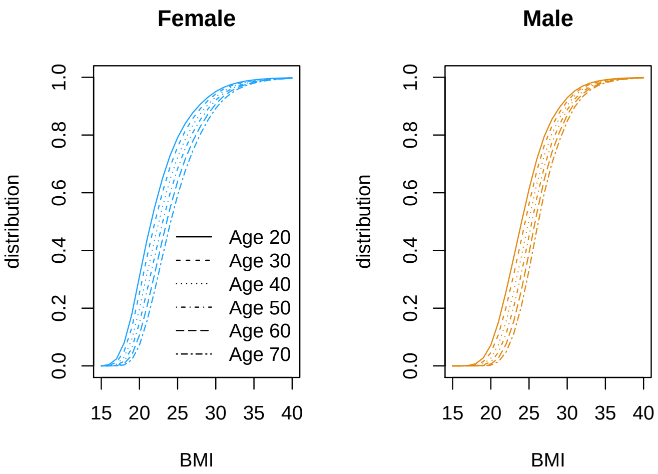
Stratified Linear Transformation Models

Include linear effects of age but still stratify wrt sex

$$\mathbb{P}(Y \le y \mid \text{sex}, \text{age}) = \\ \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y) \otimes \boldsymbol{b}(\text{sex})^\top \boldsymbol{\vartheta} + \beta(\text{sex})\text{age}))$$

```
SGB12$age <- as.double(SGB12$age)
(cic <- confint(m1Lsa <- Lehmann(bmi | sex ~ age,
                                 data = SGB12, weights = wght)))

##          2.5 %     97.5 %
## age 0.01612322 0.01623568

(cii <- confint(m2Lsa <- Lehmann(bmi | sex ~ age:sex,
                                 data = SGB12, weights = wght)))

##                    2.5 %     97.5 %
## age:sexFemale 0.01542244 0.01558351
## age:sexMale   0.01674450 0.01690156
```

Derived Conditional Distributions



Towards Distribution Regression

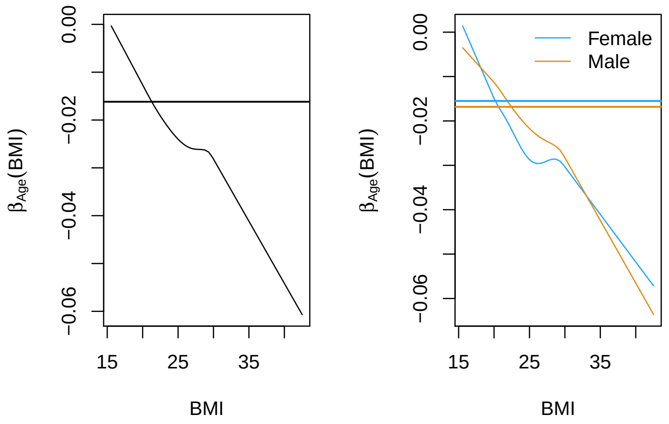
Include response-varying effect of age

$$\mathbb{P}(Y \le y \mid \text{sex}, \text{age}) = \\ \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y) \otimes \boldsymbol{b}(\text{sex})^\top \boldsymbol{\vartheta} + \beta(y \mid \text{sex})\text{age}))$$

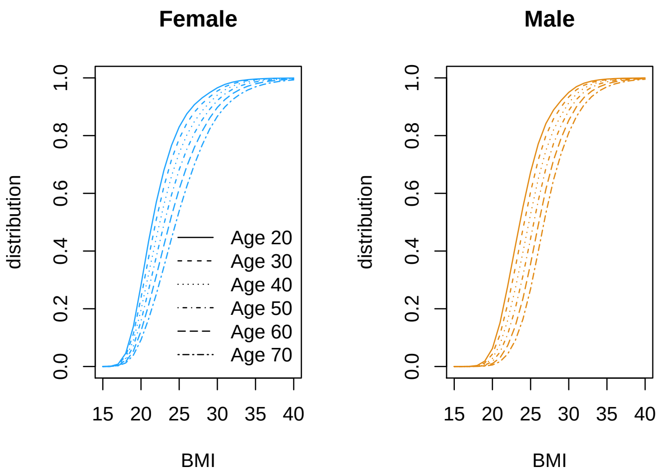
```
m3Lsa <- Lehmann(bmi | sex + age ~ 1, data = SGB12, weights = wght)
m4Lsa <- Lehmann(bmi | sex * age ~ 1, data = SGB12, weights = wght)
```

Towards Distribution Regression

Look at linear age effect as a function of BMI



Derived Conditional Distributions



Conditional Transformation Model

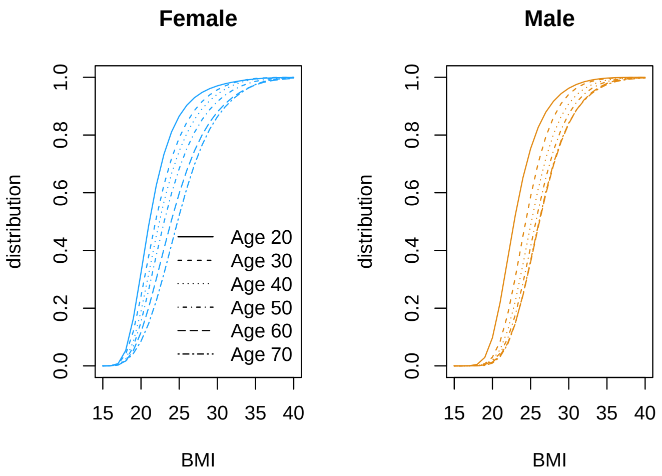
Transformation is a smooth bivariate of BMI and age, separately for females/males.

$$\mathbb{P}(Y \le y \mid \text{sex}, \text{age}) = \text{loglog}^{-1}(\boldsymbol{a}_{\text{Bs},P-1}(y) \otimes \boldsymbol{a}_{\text{Bs},P-1}(\text{age}) \otimes \boldsymbol{b}(\text{sex})^\top \vartheta))$$

Use package **mlt** directly

```
vbmi <- numeric_var("bmi", bounds = c(0, Inf), support = c(17, 35))
vage <- numeric_var("age", bounds = c(0, Inf), support = c(18, 80))
bbmi <- Bernstein_basis(vbmi, order = 5, ui = "increasing")
bage <- Bernstein_basis(vage, order = 5)
bsex <- as.basis(~ 0 + sex, data = SGB12)
m <- ctm(response = bbmi, interacting = b(age = bage, sex = bsex),
         todistr = "MaxExtrVal")
logLik(m5Lsa <- mlt(m, data = SGB12, weights = SGB12$wght))

## 'log Lik.' -13423911 (df=72)
```

Derived Conditional Distributions



Sampling from Models

Implement parametric bootstrap by sampling from fully specified models for conditional distributions

```
simulate(m5Lsa, newdata = nda, nsim = 4)

## [[1]]
##  [1] 22.85672 28.90697 21.57849 24.40836 26.67115 25.33570 20.69618
##  [8] 32.57490 20.57740 24.82212 24.75750 26.20107
##
## [[2]]
##  [1] 22.70362  20.20712  18.19583  23.21027  19.34605  21.61570  22.15208
##  [8] 24.31731  23.52776  25.89791  (35, Inf] 23.01889
##
## [[3]]
##  [1] 19.82946 27.07680 24.44095 29.33067 20.80674 30.22515 21.00202
##  [8] 28.31870 18.90617 25.11439 28.11486 26.64654
##
## [[4]]
##  [1] 23.05378 19.74204 21.62640 27.16635 20.36040 21.79553 24.87598
##  [8] 27.18179 23.42387 25.39223 29.69830 21.67717
```

Likelihood Inference

Model Estimation (Discrete)

Model: $y \in \{y_1, \dots, y_K\}$, $\boldsymbol{x} \in \mathbb{R}^Q$

$$\mathbb{P}(Y \le y_k \mid \boldsymbol{X} = \boldsymbol{x}) = F_Z(\vartheta_k - \boldsymbol{x}^\top \boldsymbol{\beta})$$

Observe *datum* $(y, \boldsymbol{x})$ and evaluate density (=probability) for GIVEN parameters $\boldsymbol{\vartheta} = (\vartheta_1, \dots, \vartheta_{K-1}), \boldsymbol{\beta}$

$$L((y, \boldsymbol{x})) = \mathbb{P}(Y = y \mid \boldsymbol{X} = \boldsymbol{x}) = f_Y(y \mid \boldsymbol{x}) = \begin{cases} F_Z(\vartheta_k - \boldsymbol{x}^\top \boldsymbol{\beta}) & - & 0 & k = 1 \\ F_Z(\vartheta_k - \boldsymbol{x}^\top \boldsymbol{\beta}) & - & F_Z(\vartheta_{k-1} - \boldsymbol{x}^\top \boldsymbol{\beta}) & 1 < k < K \\ 1 & - & F_Z(\vartheta_{k-1} - \boldsymbol{x}^\top \boldsymbol{\beta}) & k = K \end{cases}$$

Model Estimation (Discrete)

Observe *data* $(y, \boldsymbol{x})_i, i = 1, \dots, N$ and assume $(y, \boldsymbol{x})_i$ and $(y, \boldsymbol{x})_j$ are independent $\forall i \neq j$

$$L(\boldsymbol{\vartheta}, \boldsymbol{\beta}) = \prod_{i=1}^{N} L((y, \boldsymbol{x})_i)$$

is the probability of observing the data GIVEN $\boldsymbol{\vartheta}, \boldsymbol{\beta}$.

$$\hat{\boldsymbol{\vartheta}}_N, \hat{\boldsymbol{\beta}}_N = \operatorname*{arg\,max}_{\boldsymbol{\vartheta} \in \mathbb{R}^{K-1}, \boldsymbol{\beta} \in \mathbb{R}^Q} \log(L(\boldsymbol{\vartheta}, \boldsymbol{\beta})) \quad \text{st. } \vartheta_k < \vartheta_{k+1}, 1 \le k < K$$

with

$$\log(L(\boldsymbol{\vartheta}, \boldsymbol{\beta})) = \ell(\boldsymbol{\vartheta}, \boldsymbol{\beta}) = \sum_{i=1}^{N} \ell_i(\boldsymbol{\vartheta}, \boldsymbol{\beta})$$

Likelihood Function (Continuous)

Continuous RVs with $\Xi = \mathbb{R}$.

We *always* observe intervals $(\underline{y}, \bar{y}] \subset \mathbb{R}$

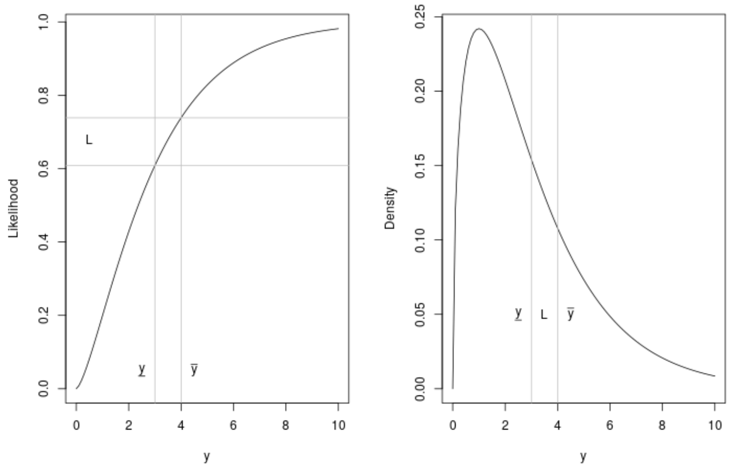
For datum $((\underline{y}, \bar{y}], \boldsymbol{x})$ evaluate *probability*

$$\begin{aligned}\mathbb{P}(\underline{y} < Y \le \bar{y} \mid \boldsymbol{X} = \boldsymbol{x}) &= F_Z(\boldsymbol{a}(\bar{y})^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta}) - \\ &\qquad F_Z(\boldsymbol{a}(\underline{y})^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta})\end{aligned}$$

Log-likelihood contribution of $((\underline{y}, \bar{y}], \boldsymbol{x})_i$ is

$$\ell_i(\boldsymbol{\vartheta}, \boldsymbol{\beta}) = \log(F_Z(\boldsymbol{a}(\bar{y})^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta}) - F_Z(\boldsymbol{a}(\underline{y})^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta}))$$

Likelihood Function (Continuous)



Likelihood Function (Continuous)

If measurement is precise (length of $(\underline{y}, \bar{y}]$ short), *APPROXIMATE* likelihood by density evaluated at $y = \underline{y} + (\bar{y} - \underline{y})/2$

$$\ell(\boldsymbol{\vartheta}, \boldsymbol{\beta}) \approx \log(f_Y(y \mid \boldsymbol{X} = \boldsymbol{x}))$$

Problem: What is f_Y?

$$f_Y(y \mid \boldsymbol{X} = \boldsymbol{x}) = \frac{\partial F_Y(y \mid \boldsymbol{X} = \boldsymbol{x})}{\partial y}$$

Likelihood Function (Continuous)

$$f_{Y \mid \boldsymbol{X} = \boldsymbol{x}}(y \mid \boldsymbol{X} = \boldsymbol{x}) = f_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta}) \boldsymbol{a}'(y)^\top \boldsymbol{\vartheta}$$

$(y_i, \boldsymbol{x}_i)$ for $i = 1, \dots, N$ with log-likelihood contribution

$$\ell_i(\boldsymbol{\vartheta}, \beta) = \log(f_Z(\boldsymbol{a}(y_i)^\top \boldsymbol{\vartheta} - \boldsymbol{x}_i^\top \beta)) + \log(\boldsymbol{a}'(y_i)^\top \boldsymbol{\vartheta})$$

Score Function

$$\boldsymbol{s}(\boldsymbol{\vartheta}, \boldsymbol{\beta}) = \frac{\partial \ell(\boldsymbol{\vartheta}, \boldsymbol{\beta})}{\partial(\boldsymbol{\vartheta}, \boldsymbol{\beta})} = \sum_{i=1}^{N} \boldsymbol{s}_i(\boldsymbol{\vartheta}, \boldsymbol{\beta}) = \sum_{i=1}^{N} \frac{\partial \ell_i(\boldsymbol{\vartheta}, \boldsymbol{\beta})}{\partial(\boldsymbol{\vartheta}, \boldsymbol{\beta})}$$

Under some regularity conditions we have

$$\boldsymbol{s}(\hat{\boldsymbol{\vartheta}}_N, \hat{\boldsymbol{\beta}}_N) = 0$$

Likelihood Inference

```
logLik(m1L)

## 'log Lik.' -13677664 (df=8)

logLik(m1L, newdata = SGB12[1:10,])

## 'log Lik.' -25.10254 (df=NULL)

(cf <- coef(as.mlt(m1L)))

##     Bs1(bmi)     Bs2(bmi)     Bs3(bmi)     Bs4(bmi)     Bs5(bmi)
## -0.581819812  0.006656806  0.474723543  0.835574128  1.680898343
##     Bs6(bmi)     Bs7(bmi)      sexMale
##  2.042658001  2.681051379  0.735121200

cf["sexMale"] <- 0
logLik(m1L, parm = cf)

## 'log Lik.' -14221888 (df=8)

logLik(m1L, w = runif(nrow(SGB12)))

## 'log Lik.' -22026.91 (df=8)
```

Likelihood Inference

```
head(vcov(as.mlt(m1L)), 4)

##               Bs1(bmi)      Bs2(bmi)      Bs3(bmi)      Bs4(bmi)
## Bs1(bmi)  4.643010e-07  5.833346e-07 -1.882011e-07  8.964253e-07
## Bs2(bmi)  5.833346e-07  1.045332e-06 -7.597083e-07  1.704403e-06
## Bs3(bmi) -1.882011e-07 -7.597083e-07  8.188317e-06 -9.713507e-06
## Bs4(bmi)  8.964253e-07  1.704403e-06 -9.713507e-06  1.928851e-05
##               Bs5(bmi)      Bs6(bmi)     Bs7(bmi)      sexMale
## Bs1(bmi)  5.445500e-08  3.590977e-07 3.461403e-07 3.780447e-07
## Bs2(bmi) -1.815578e-07  4.994011e-07 4.712803e-07 4.843515e-07
## Bs3(bmi)  6.476553e-06 -1.245604e-07 1.615585e-07 5.384677e-07
## Bs4(bmi) -1.190608e-05  2.332565e-06 1.611714e-06 5.565496e-07

head(estfun(m1L), 4)

##          Bs1(bmi)      Bs2(bmi)     Bs3(bmi)   Bs4(bmi)  Bs5(bmi)
## [1,] 2.221412e+02 -7.572542e+01 -131.7384744 -47.010338 -7.612575
## [2,] 3.335931e-05  4.354623e-03    0.2269266   5.889196 75.645341
## [3,] 1.252692e-02  3.697307e-01    4.3227356  24.758795 67.042383
## [4,] 4.673555e+01 -1.744076e+02    0.0000000   0.000000  0.000000
##         Bs6(bmi)     Bs7(bmi)   sexMale
## [1,]  -0.5961512  -0.01838774    0.0000
## [2,] 372.4530123 -210.28438046 -243.9345
## [3,]  54.0033926  -53.30114152    0.0000
## [4,]   0.0000000    0.00000000    0.0000
```

Likelihood Inference

```
summary(m1L)

##
##   Lehmann-alternative Linear Regression Model
##
## Call:
## Lehmann(formula = bmi ~ sex, data = SGB12, weights = wght)
##
## Coefficients:
##         Estimate Std. Error z value Pr(>|z|)
## sexMale 0.735121   0.000935   786.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood:
##  -13677664 (df = 8)
## Likelihood-ratio Test: Chisq = 611632.1 on 1 degrees of freedom; p = < 2.2e-16
```

Model Comparison and Diagnostics

Model Comparison

How can we compare two (or more) models for $\mathbb{P}(Y \le y \mid \boldsymbol{X} = \boldsymbol{x})$?

Idea: Compare *out-of-sample* log-likelihoods

- Estimate model parameters (use maximum likelihood or whatever) for observations $(y, \boldsymbol{x})_i, i = 1, \dots, N$, this gives $\hat{\boldsymbol{\vartheta}}_N, \hat{\boldsymbol{\beta}}_N$
- Obtain $\tilde{N}$ *independent* new observations $(y, \boldsymbol{x})_i, i = N + 1, \dots, N + \tilde{N}$
- Compute out-of-sample log-likelihood

$$\tilde{\ell}(\hat{\boldsymbol{\vartheta}}_N, \hat{\boldsymbol{\beta}}_N) = \sum_{i=N+1}^{N+\tilde{N}} \ell_i(\hat{\boldsymbol{\vartheta}}_N, \hat{\boldsymbol{\beta}}_N)$$

- also known as proper “log-score” in the scoring rules literature

---

Looking at Distributions

```
nd <- expand.grid(sex = factor(c("Female", "Male")), bmi = c(18.5, 25, 30))
predict(m1Ls, newdata = nd, type = "distribution")

##          1          2          3          4          5          6
## 0.04939904 0.00345106 0.70015846 0.48089956 0.92489357 0.89386830

predict(m1Ls, newdata = nd, type = "density")

##           1           2           3           4           5           6
## 0.055768432 0.005777055 0.073397436 0.126520308 0.024869211 0.042851469

predict(m1Ls, newdata = nd, type = "hazard")

##           1           2           3           4           5           6
## 0.058666501 0.005797061 0.244787416 0.243729916 0.331119615 0.403757484

predict(m1Ls, newdata = nd[1:2,], type = "quantile", prob = 1:3 / 4)

##
## prob        [,1]     [,2]
##    0.25 20.60552 23.08066
##     0.5 22.80669 25.15307
##    0.75 25.74558 27.50498
```

Model Diagnostics: PIT

Probability Integral Transform (PIT)

$$U_i = \mathbb{P}(Y \le y_i \mid \boldsymbol{X} = \boldsymbol{x}_i) \sim \text{U}[0, 1]$$

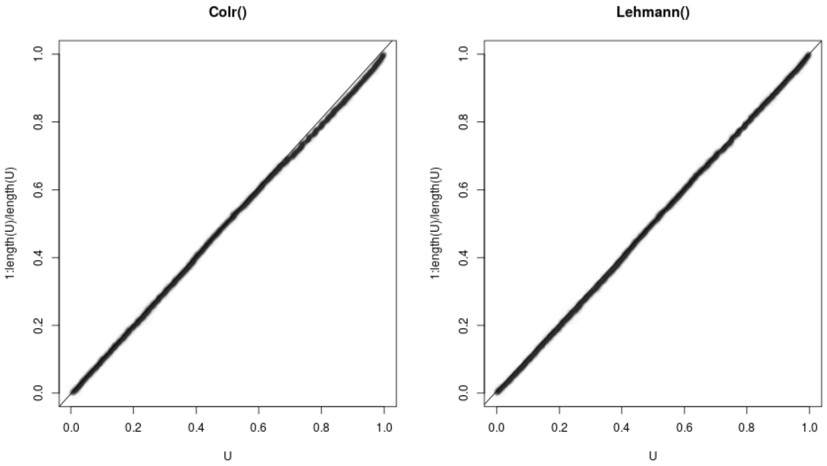
Idea: Check

$$\hat{U}_i = \hat{\mathbb{P}}(Y \le y_i \mid \boldsymbol{X} = \boldsymbol{x}_i) = F_Z(\boldsymbol{a}(y_i)^\top \hat{\boldsymbol{\vartheta}} - \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}}) \sim \text{U}[0, 1]$$

or equivalently

$$\hat{Z}_i = \boldsymbol{a}(y_i)^\top \hat{\boldsymbol{\vartheta}} - \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}} \sim \mathbb{P}_Z$$

BMI Quantile-Quantile Plots



Residual Analysis

Model

$$F_{Y|\boldsymbol{X}=\boldsymbol{x}}(y) = F_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta} - \alpha), \quad \alpha = 0$$

Idea: If this simple model is correct $\Rightarrow \alpha(\boldsymbol{x}) \equiv 0$ in the more complex model

$$F_{Y|\boldsymbol{X}=\boldsymbol{x}}(y) = F_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} - \boldsymbol{x}^\top \boldsymbol{\beta} - \alpha(\boldsymbol{x}))$$

Estimate simple model ($\alpha = 0$) and look for deviations

Residual Analysis

What is a residual?

Log-Likelihood contribution

$$\ell_i((\boldsymbol{\vartheta}, \boldsymbol{\beta}, \alpha = 0))$$

Residual := Score contribution wrt α

$$r_i = \left.\frac{\partial \ell_i((\boldsymbol{\vartheta}, \boldsymbol{\beta}, \alpha))}{\partial \alpha}\right|_{\alpha=0}$$

Any association between r_i and $\boldsymbol{x}_i$?

Example: Score Test for Comparing Two Groups

Model:

$$\begin{aligned} \mathbb{P}(Y \le y \mid \text{placebo}) &= \text{expit}(h(y)) \\ \mathbb{P}(Y \le y \mid \text{treatment}) &= \text{expit}(h(y) - \beta) \end{aligned}$$

$H_0 : \beta = 0$ vs. log-odds ratio alternatives

Observe $(y, \boldsymbol{x})_i, i = 1, \dots, N$ (independent etc)

Under H_0 (!!!), estimate cumulative distribution function

$$F_Y(y) = \mathbb{P}(Y \le y)$$

from the whole sample

Example: Score Test for Comparing Two Groups

Maybe very simple by ECDF

$$\hat{F}_{Y,N}(y_i) = (N+1)^{-1} \sum_{j=1}^{N} \mathbb{1}(y_j \leq y_i) = (N+1)^{-1} R_i$$

where R_i is the rank of the ith response value in the whole sample

Then: $\hat{h}(y_i) = \text{logit}((N+1)^{-1} R_i)$

Example: Score Test for Comparing Two Groups

Plug-in $\hat{h}(y_i)$ and compute score wrt $\alpha \equiv 0$

$$r_i = \left.\frac{\partial \ell_i(\hat{h}(y_i), \alpha)}{\partial \alpha}\right|_{\alpha=0} = 1 - 2R_i/(N+1)$$

Use “correlation” between score and treatment as test statistic:

$$\sum_{i=1}^{N} r_i \mathbb{1}(\boldsymbol{x}_i = \text{treatment}) \cong \sum_{i=1}^{N} R_i \mathbb{1}(\boldsymbol{x}_i = \text{treatment}) = W$$

Example: Score Test for Comparing Two Groups

Plug-in $\hat{h}(y_i)$ and compute score wrt $\alpha \equiv 0$

$$r_i = \left.\frac{\partial \ell_i(\hat{h}(y_i), \alpha)}{\partial \alpha}\right|_{\alpha=0} = 1 - 2R_i/(N+1)$$

Use “correlation” between score and treatment as test statistic:

$$\sum_{i=1}^{N} r_i \mathbb{1}(\boldsymbol{x}_i = \text{treatment}) \cong \sum_{i=1}^{N} R_i \mathbb{1}(\boldsymbol{x}_i = \text{treatment}) = W$$

Oups: Wilcoxon-Mann-Whitney-Rank-Sum Test

Example: Score Test for Comparing Two Groups

The parameterisation $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$ opens a whole new world to conditional inference

```r
### proportional odds alternatives (Wilcoxon)
resid(Colr(...))
### proportional hazards alternatives (Log-rank)
resid(Coxph(...))
### shift alternatives (van der Waerden)
resid(BoxCox(...))
```

potentially under random censoring, truncation, stratification, or covariate adjustment. Use `coin::independence_test()` to compute test statistics and permutation distributions.

Illustration: Disease-free Survival Time

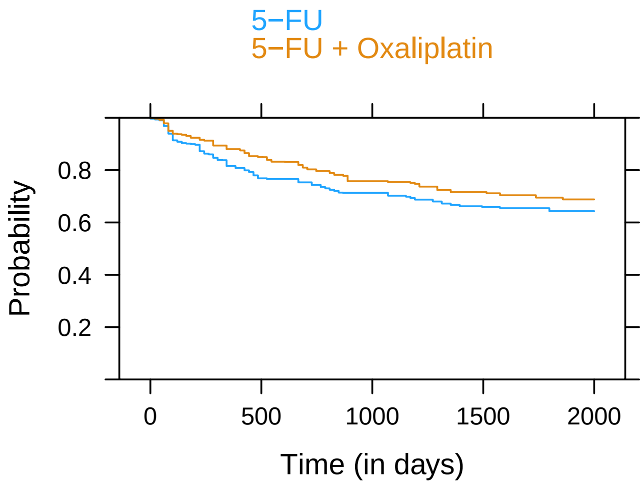
CAO/AIO/ARO-04 rectal cancer trial

Primary outcome is time to first occurance of

- non-radical surgery of the primary tumour (R2),
- locoregional recurrence after R0/1 resection,
- metastatic disease or progression,
- or death from any cause

R2 and death are “exact” dates; recurrence of progression is in fact interval-censored and depends on follow-up schedule!

Illustration: Disease-free Survival Time



Turnbull estimator

Illustration: Disease-free Survival Time

$$\mathbb{P}(Y > y \mid \text{treatment}) = \exp(-\exp(\boldsymbol{a}(\log(y))^\top \boldsymbol{\vartheta} + \beta I(\text{5-FU + Ox})))$$

Fully parametric Cox model for intervall-censored survival times

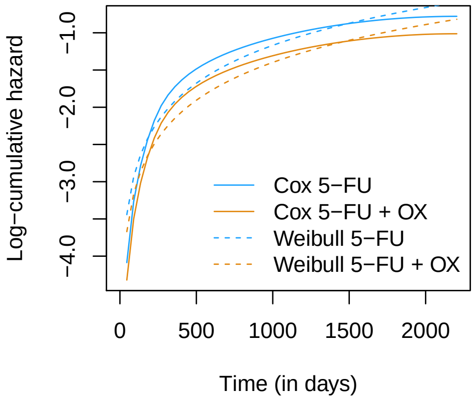
```
logLik(m1Cx <- Coxph(iDFS ~ randarm, data = CAOsurv, log_first = TRUE))

## 'log Lik.' -2255.739 (df=8)

exp(c(coef(m1Cx), confint(m1Cx)))

## randarm5-FU + Oxaliplatin
##                 0.7913836                 0.6423038
##
##                 0.9750651
```

Cox vs. Weibull

$$\mathbb{P}(Y > y \mid \text{treatment}) = \exp(-\exp(\vartheta_1 + \vartheta_2 \log(y) + \beta I(\text{5-FU + Ox})))$$

```
logLik(mC <- Coxph(iDFS ~ randarm, data = CAOsurv, log_first = TRUE, order = 1))

## 'log Lik.' -2281.171 (df=3)

logLik(mS <- Survreg(iDFS ~ randarm, data = CAOsurv))

## 'log Lik.' -2281.171 (df=3)

logLik(survival::survreg(iDFS ~ randarm, data = CAOsurv))

## 'log Lik.' -2281.171 (df=3)
```

Cox vs. Weibull



Exact Conditional Inference

Log-rank test for interval-censored observations

```
CAOsurv$sc <- resid(Coxph(iDFS ~ 1, data = CAOsurv, log_first = TRUE))
library("coin")
pvalue(independence_test(sc ~ randarm, data = CAOsurv,
                         distribution = approximate(1e6)))

## [1] 0.028438
## 99 percent confidence interval:
##  0.02801159 0.02886890

library("interval")
ictest(iDFS ~ randarm, data = CAOsurv,
       method = "exact.mc")$p.values["p.twosided"]

## p.twosided
##      0.028
```

Exact Conditional Inference

OK, but what about stratification?

```
CAOsurv$stra <- with(CAOsurv, interaction(strat_t, strat_n))
CAOsurv$scs <- resid(Coxph(iDFS | 0 + stra ~ 1,
                     data = CAOsurv, log_first = TRUE))
independence_test(scs ~ randarm | stra, data = CAOsurv,
                  distribution = approximate(1e6))

##
##  Approximative General Independence Test
##
## data:  scs by
##   randarm (5-FU, 5-FU + Oxaliplatin)
##   stratified by stra
## Z = -2.1736, p-value = 0.02937
## alternative hypothesis: two.sided
```

Assessing Model Deviations

Good, we go with

```
(m2Cx <- Coxph(iDFS | 0 + stra ~ randarm, data = CAOsurv, log_first = TRUE))

##
##   Parametric Linear Cox Regression Model
##
## Call:
## Coxph(formula = iDFS | 0 + stra ~ randarm, data = CAOsurv, log_first = TRUE)
##
## Coefficients:
## randarm5-FU + Oxaliplatin
##                -0.2322697
##
## Log-Likelihood:
##  -2232.476 (df = 29)
```

Q:

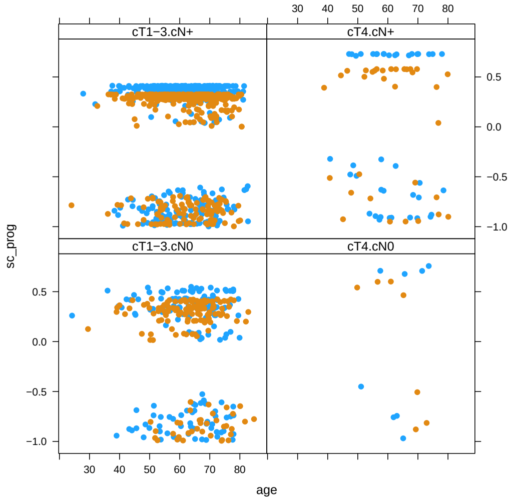
- Is there a prognostic effect of age?
- Is there a predictive effect of age?

Assessing Age Effects

$$\mathbb{P}(Y > y \mid \text{treatment, age}) = \exp(-\exp(\boldsymbol{a}(\log(y))^\top \boldsymbol{\vartheta}(\text{stra}) + \beta I(\text{5-FU + Ox}) + \mu(\text{age})))$$

```
CAOsurv$sc_prog <- resid(m2Cx)
maxstat_test(sc_prog ~ age | stra, data = CAOsurv)

##
##  Asymptotic Generalized Maximally Selected Statistics
##
## data:  sc_prog by age
##   stratified by stra
## maxT = 2.4814, p-value = 0.1622
## alternative hypothesis: two.sided
## sample estimates:
##   "best" cutpoint: <= 69.9
```

Assessing Age Effects



Assessing Age Effects

$$\mathbb{P}(Y > y \mid \text{treatment, age}) = \exp(-\exp(\vartheta_1 + \vartheta_2 \log(y) + \beta(\text{age}) I(\text{5-FU + Ox})))$$

```
CAOsurv$sc_pred <- estfun(m2Cx)[, "randarm5-FU + Oxaliplatin"]
maxstat_test(sc_pred ~ age | stra, data = CAOsurv)

##
##  Asymptotic Generalized Maximally Selected Statistics
##
## data:  sc_pred by age
##   stratified by stra
## maxT = 3.577, p-value = 0.00757
## alternative hypothesis: two.sided
## sample estimates:
##   "best" cutpoint: <= 68.9
```

Modelling Age Effects

```
CAOsurv$age69 <- with(CAOsurv, cut(age, breaks = c(0, 69, 100)))
coef(m3Cx <- Coxph(iDFS | 0 + stra ~ randarm*age69, data = CAOsurv,
                   log_first = TRUE))

##               randarm5-FU + Oxaliplatin
##                             -0.44015565
##                           age69(69,100]
##                             -0.07038779
## randarm5-FU + Oxaliplatin:age69(69,100]
##                              0.65114475

K <- diag(3)[-2,]
K[2,1] <- 1
rownames(K) <- paste(names(coef(m3Cx))[1], c(" <= 69", " > 69"))
library("multcomp")
round(exp(confint(glht(m3Cx, linfct = K))$confint), 4)

##                                  Estimate    lwr    upr
## randarm5-FU + Oxaliplatin  <= 69   0.6439 0.4793 0.8652
## randarm5-FU + Oxaliplatin  > 69    1.2349 0.8140 1.8735
## attr(,"conf.level")
## [1] 0.95
## attr(,"calpha")
## [1] 2.236422
```

Statistical Learning of Transformations

Core Idea

Start with a simple model

$$\mathbb{P}(Y \le y \mid \boldsymbol{x}) = F_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta} + \alpha + \boldsymbol{x}^\top \beta), \alpha = 0$$

and try to find better models of the form

$$\mathbb{P}(Y \le y \mid \boldsymbol{x}, \boldsymbol{z}) = F_Z(\boldsymbol{a}(y)^\top \boldsymbol{\vartheta}(\boldsymbol{z}) + \alpha(\boldsymbol{z}) + \boldsymbol{x}^\top \boldsymbol{\vartheta}(\boldsymbol{z}))$$

by modelling (some elements of) the score vector

$$\frac{\partial \ell_i(\boldsymbol{\vartheta}, \alpha, \boldsymbol{\beta})}{\partial(\boldsymbol{\vartheta}, \alpha, \boldsymbol{\beta})} \sim \boldsymbol{z}_i$$

depending on additional predictor/partitioning variables $\boldsymbol{z}$ (NOTE: $\boldsymbol{z}$ has nothing to do with F_Z, I'm just a bit lazy here).

Transformation Trees

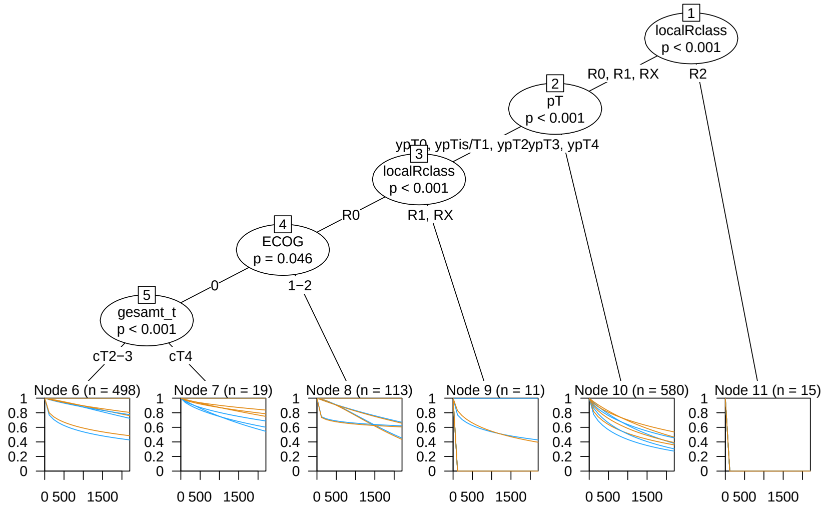
Start with

$$\mathbb{P}(Y > y \mid \text{treatment}) = \exp(-\exp(\vartheta_1(\text{stra}, \boldsymbol{z}) + \vartheta_2(\text{stra}, \boldsymbol{z}) \log(y) + \beta(\boldsymbol{z}) I(\text{5-FU + Ox})))$$

Look for changes in $\vartheta_1(\text{stra})$, $\vartheta_2(\text{stra})$ and β using model-based recursive partitioning (beefed-up `maxstat_test()`):

```
library("trtf")
fm <- iDFS | stra + randarm ~ age + geschlecht + ECOG + bentf + gesamt_t +
      gesamt_n_col + pT + localRclass + path_stad + op_meth
m <- Survreg(iDFS | 0 + stra ~ randarm, data = CAOsurv)
tr <- trafotree(m, formula = fm, data = CAOsurv)
logLik(tr)

## 'log Lik.' -2126.164 (df=54)
```

Transformation Trees



Random Forests

$(y_i, \mathbf{z}_i)$ with unconditional log-likelihood contribution ℓ_i

$$\hat{\vartheta}(\mathbf{z}) = \operatorname*{arg\,max}_{\vartheta} \sum_{i=1}^{N} w_i(\mathbf{z}) \ell_i(\vartheta)$$

is called “adaptive local maximum likelihood estimator”.

Random forests can be used to define “nearest neighbor” weights w_i.

Transformation Forests

Weibull model

$$\mathbb{P}(Y > y \mid \text{treatment}) =$$
$$\exp(-\exp(\vartheta_1(\text{stra}, \mathbf{z}) + \vartheta_2(\text{stra}, \mathbf{z}) \log(y) + \beta(\mathbf{z}) I(\text{5-FU + Ox}))$$

with random forest functions $\vartheta_1(\mathbf{z})$, $\vartheta_2(\mathbf{z})$, and $\beta(\mathbf{z})$. More complex models are possible.

```
tf <- traforest(m, formula = fm, data = CAOsurv,
                control = ctree_control(maxdepth = 5))
# logLik(tf)
```

Transformation Forests

We still get treatment effects on the log-hazard ratio scale!

```
coef(tr)[, "randarm5-FU + Oxaliplatin"]

##             6             7             8             9            10
##    0.16116273    0.74510677   -0.04793334 -244.21300169    0.23577070
##            11
##   -4.36836559

sapply(predict(tf, newdata = CAOsurv[1:10,], mnewdata = CAOsurv[1:10,],
        type = "coef"), function(x) x["randarm5-FU + Oxaliplatin"])

##  1.randarm5-FU + Oxaliplatin  2.randarm5-FU + Oxaliplatin
##                    0.2435321                    0.2545647
##  3.randarm5-FU + Oxaliplatin  4.randarm5-FU + Oxaliplatin
##                    0.1452141                    0.1521941
##  5.randarm5-FU + Oxaliplatin  6.randarm5-FU + Oxaliplatin
##                    0.2435321                    0.1597275
##  7.randarm5-FU + Oxaliplatin  8.randarm5-FU + Oxaliplatin
##                    0.2435321                    0.1575915
##  9.randarm5-FU + Oxaliplatin 10.randarm5-FU + Oxaliplatin
##                    0.1575915                    0.2416532
```

Transformation Boosting

Sometimes, we want a bit more structure in our models, for example

$$\mathbb{P}(Y > y \mid \text{treatment}) =$$
$$\exp(-\exp(\mathbf{a}(\log(y))^\top \boldsymbol{\vartheta}(\text{stra}) + \beta I(\text{5-FU + Ox}) + \mathbf{x}^\top \boldsymbol{\beta})))$$

where $\boldsymbol{\beta}$ shall be penalised but $\boldsymbol{\vartheta}$ and β shall not.

Use gradient boosting to update $\mathbf{x}^\top \boldsymbol{\beta}$ iteratively.

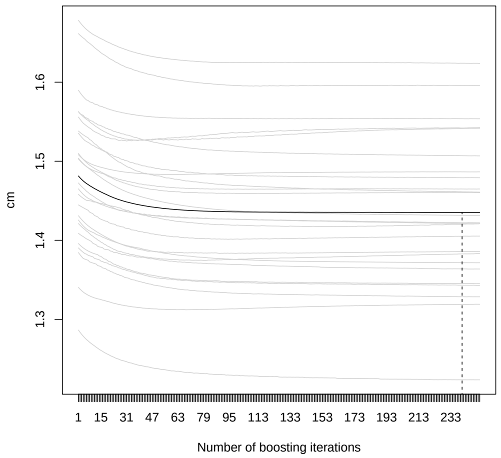
Shift Transformation Boosting

```
library("tbm")
library("mboost")
```

```
fm <- iDFS ~ bols(age, intercept = FALSE) +
             bols(geschlecht, intercept = FALSE) +
             bols(ECOG, intercept = FALSE) +
             bols(bentf, intercept = FALSE) +
             bols(gesamt_t, intercept = FALSE) +
             bols(gesamt_n_col, intercept = FALSE) +
             bols(pT, intercept = FALSE) +
             bols(localRclass, intercept = FALSE) +
             bols(path_stad, intercept = FALSE) +
             bols(op_meth, intercept = FALSE)
cc <- complete.cases(CAOsurv[, all.vars(fm)])
m <- as.mlt(Coxph(iDFS ~ randarm, data = CAOsurv[cc,], log_first = TRUE,
                  order = 1))
stm <- stmboost(m,
                formula = fm, data = CAOsurv[cc,all.vars(fm)],
                method = quote(mboost::mboost),
                control = boost_control(mstop = 250, nu = .2))
stmll <- cvrisk(stm)
```

Shift Transformation Boosting

```
plot(stmll)
```



Shift Transformation Boosting

```
table(variable.names(stm)[selected(stm)])
```

```
##
##         ECOG gesamt_n_col     gesamt_t   geschlecht  localRclass
##            6           34            4            2           27
##      op_meth    path_stad           pT
##           35           43           99
```

```
logLik(stm)
```

```
## 'log Lik.' -1608.398 (df=NULL)
```

```
nuisance(stm)
```

```
## [1] -3.0847684 -1.3245673 -0.1782098
```

Conditional Transformation Boosting

Weibull model with linear additive functions $\vartheta_1(\boldsymbol{z})$ and $\vartheta_2(\boldsymbol{z})$ (currently treatment effects not allowed).

```
m <- as.mlt(Coxph(iDFS ~ 1, data = CAOsurv[cc,], log_first = TRUE, order = 1))
ctm <- ctmboost(m,
                formula = fm, data = CAOsurv[cc,all.vars(fm)],
                method = quote(mboost::mboost),
                control = boost_control(mstop = 250, nu = .2))
ctmll <- cvrisk(ctm)
logLik(ctm[mstop(ctmll)])
```

```
## 'log Lik.' -1564.635 (df=NULL)
```

```
table(variable.names(ctm)[selected(ctm)])
```

```
##
##        bentf         ECOG gesamt_n_col   geschlecht      op_meth
##           15            3           41            8           16
##    path_stad           pT
##           18            4
```

Summary

- Generic interfaces to many regression models
- Likelihood-based inference
- Model assessment and criticism at different scales
- Interpretable parameters optionally depend on external information
- Smooth transition between simple low-parametric models and potentielly complex models defined through statistical learning procedures
- Lego look-and-feel
- Need to think a bit out-of-the-box

What's Next?

- Multivariate conditional transformation models
- Mixed transformation models
- Shift/Scale transformation models
- Use CVXR as optimiser; allows explicit penalisation (Lasso etc)
- Score-based confidence intervals for supplementing permutation p-values
- Teaching material
- Stay tuned at http://ctm.R-forge.R-project.org