Model-Based Recursive Partitioning for Stratified and Personalised Treatment Effect Estimation

Heidi Seibold

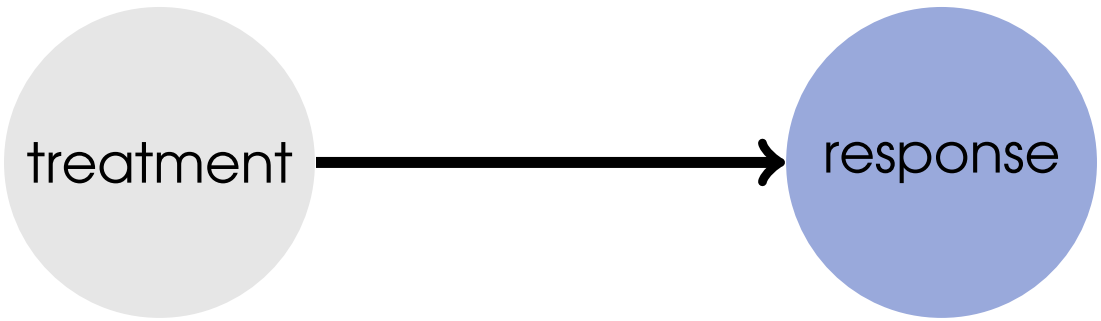
Epidemiology, Biostatistics and Prevention Institute

University of Zurich[UZH]

Joint work with Torsten Hothorn (UZH) and Achim Zeileis (UIBK), sponsored by SNF

Statistical Computing 2017

Treatment Effect




```
base_model <- model(response ~ treatment, data)
```

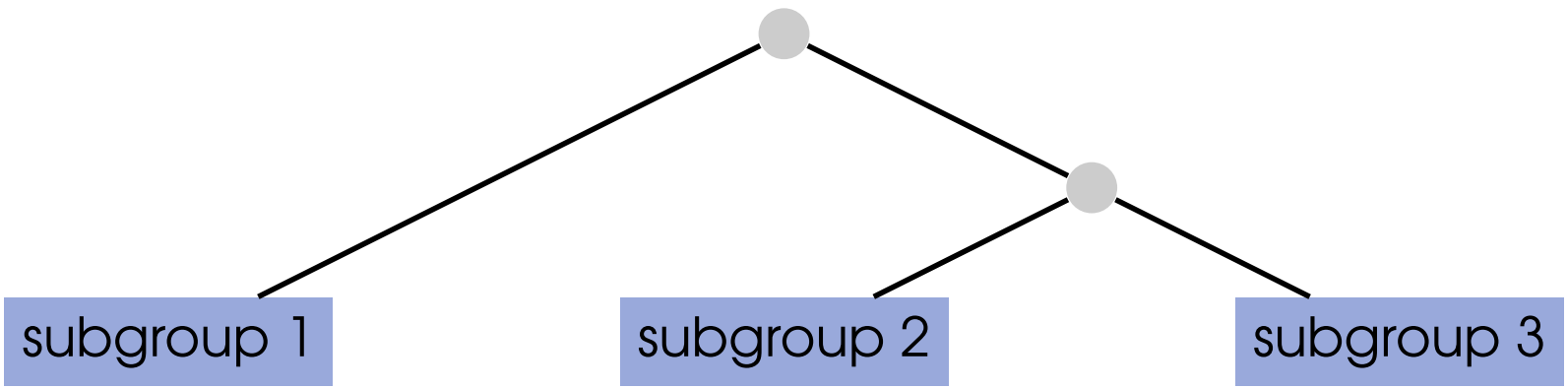
Stratified Treatment Effects

```
base_model <- model(response ~ treatment, data)


ismale <- data$gender == "male"
model_male <- model(response ~ treatment,
                    data[ismale, ])
model_female <- model(response ~ treatment,
                      data[!ismale, ])
```

Stratified Treatment Effects

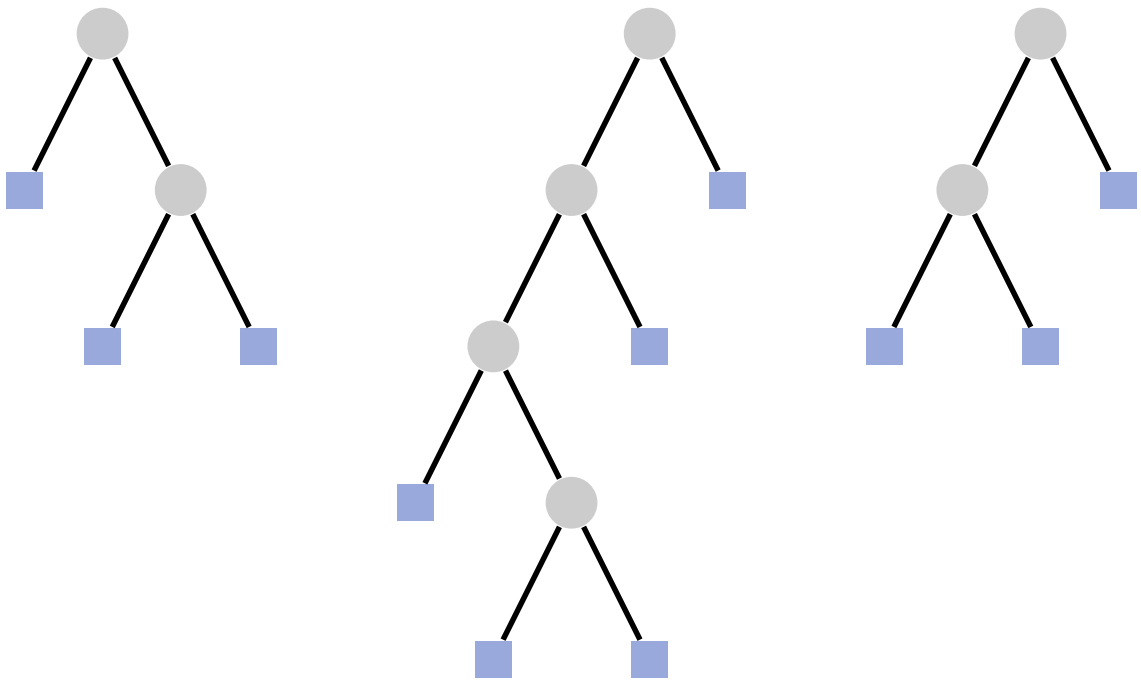
```
base_model <- model(response ~ treatment, data)


ismale <- data$gender == "male"
model_male <- model(response ~ treatment,
                    data, weights = ismale)
model_female <- model(response ~ treatment,
                      data, weights = !ismale)
```

Stratified Treatment Effects using model-based trees



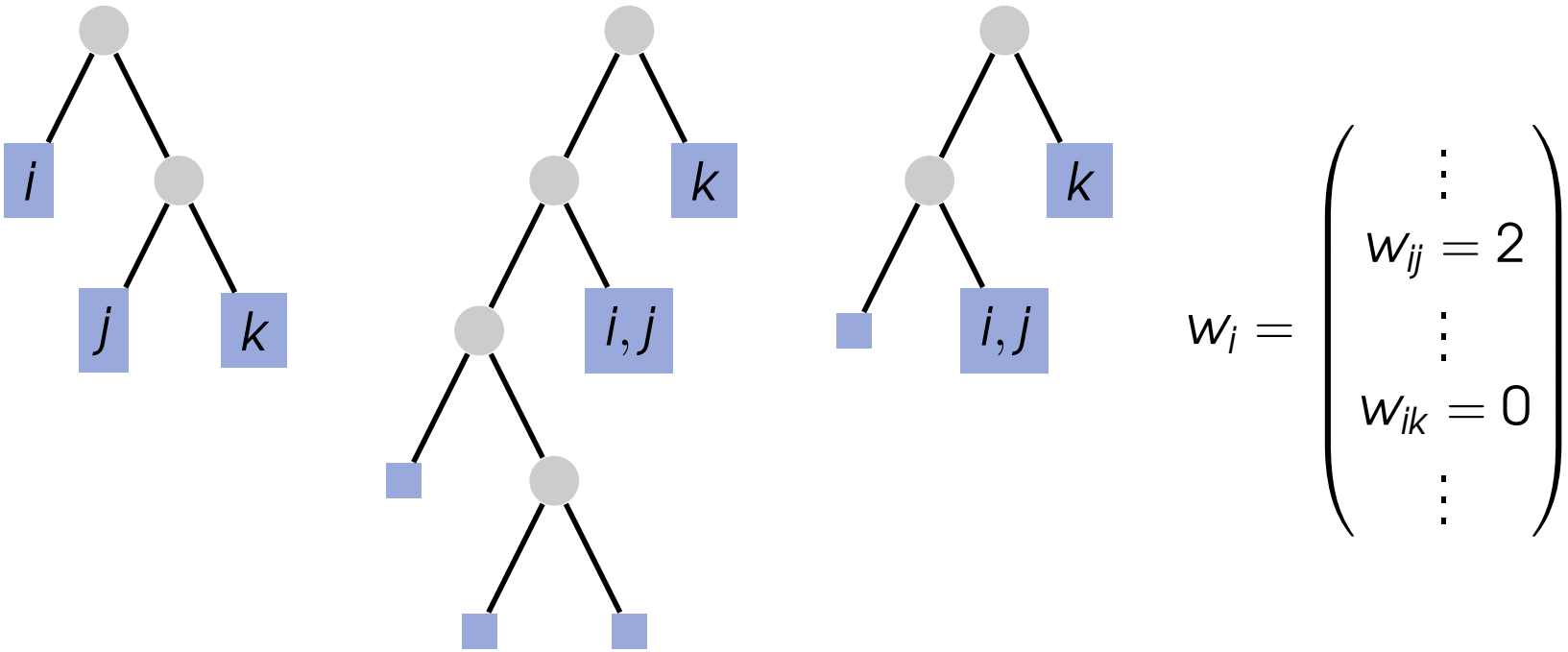

```
model_sg1 <- model(response ~ treatment, data,
                   weights = (subgroup == 1))
```

Personalised Treatment Effects using model-based forests



```
model_i <- model(response ~ treatment, data,
                 weights = w_i)
```

Personalised Treatment Effects using model-based forests




```
model_i <- model(response ~ treatment, data,
                 weights = w_i)
```

The R Package `model4you`

Base model:

```
base_model <- model(response ~ treatment, data)
```

Stratified treatment effects:

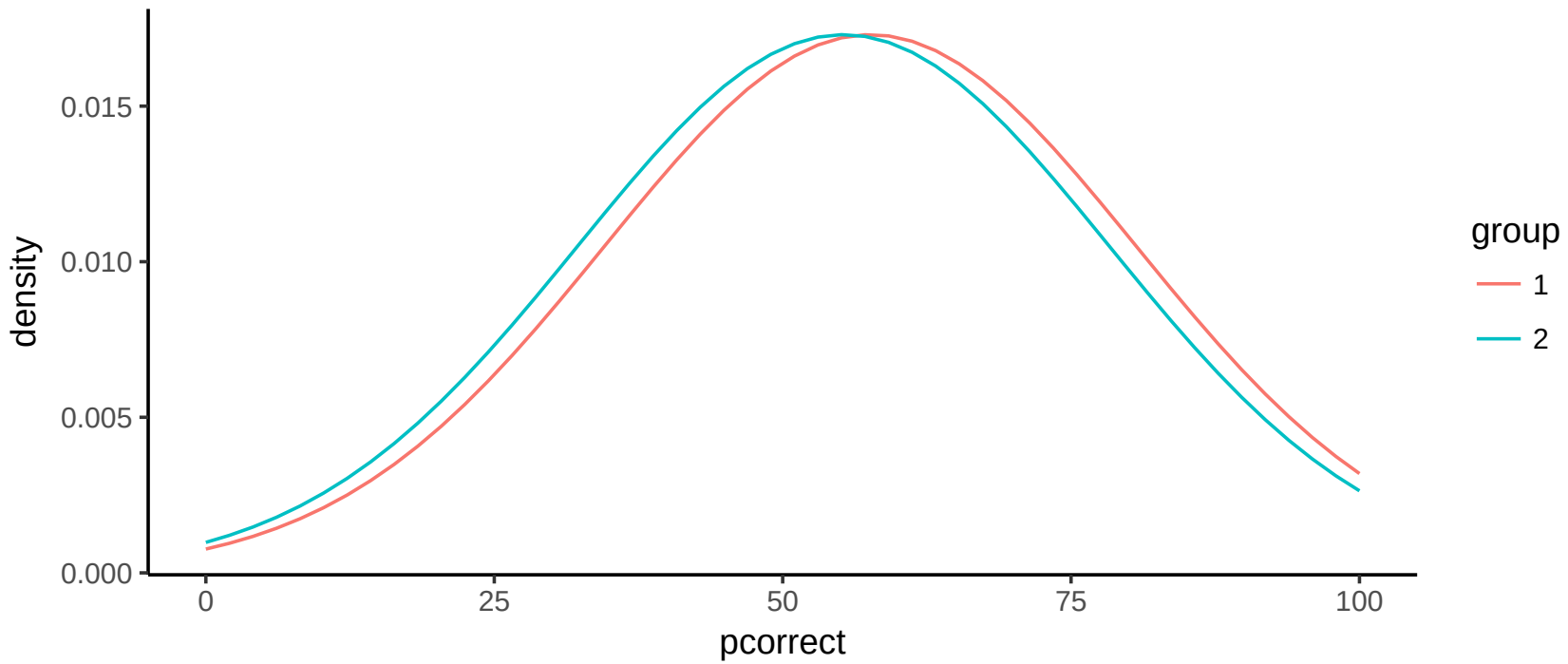
```
strat_models <- pmtree(base_model)
```

Personalised treatment effects:

```
pm_forest <- pmforest(base_model)
pers_models <- pmodel(pm_forest)
```

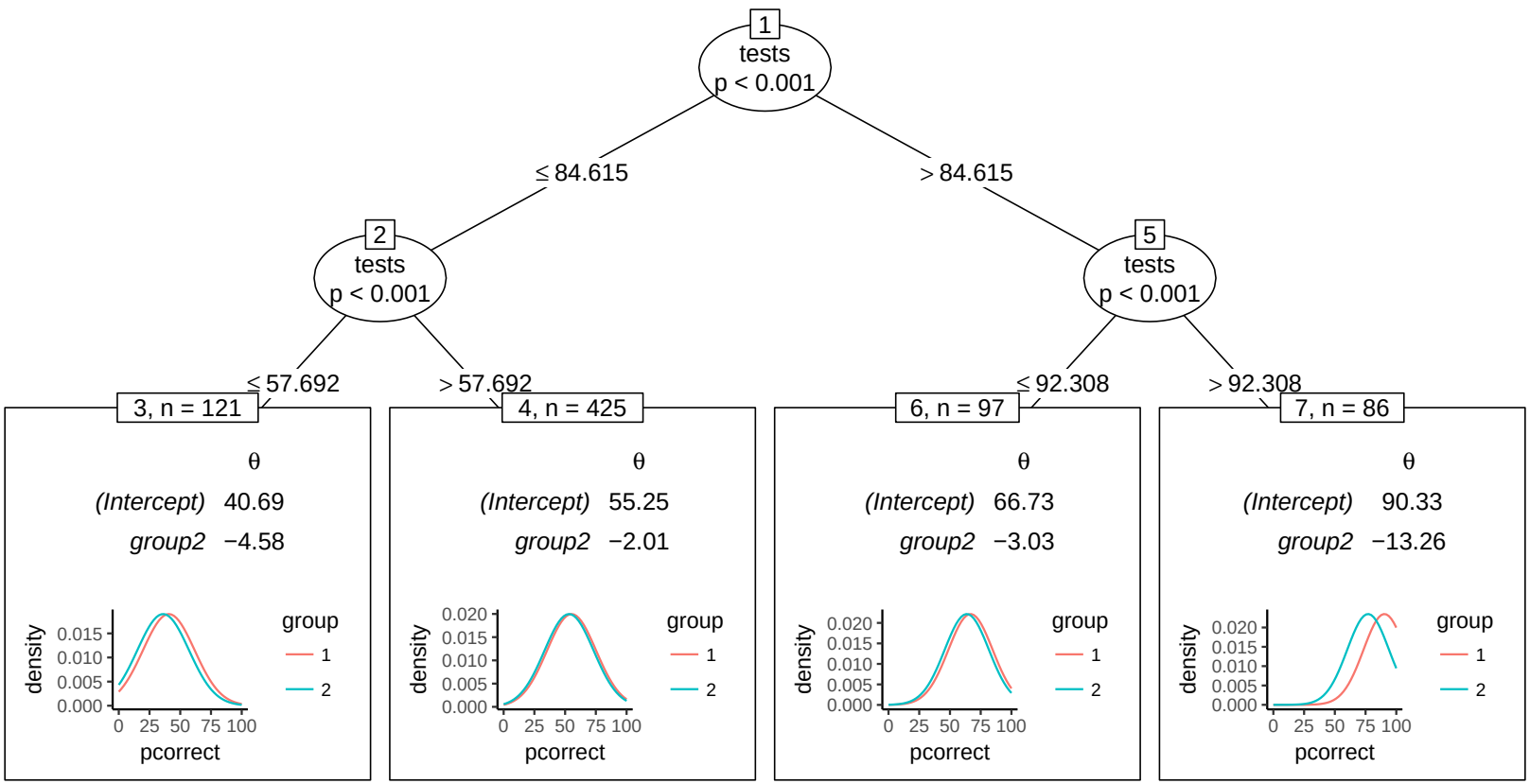
Math Exam: Base Model

```
base_model <- lm(pcorrect ~ group, data = MathExam)
lmplot(base_model)
```



Math Exam: Stratified Treatment Effects

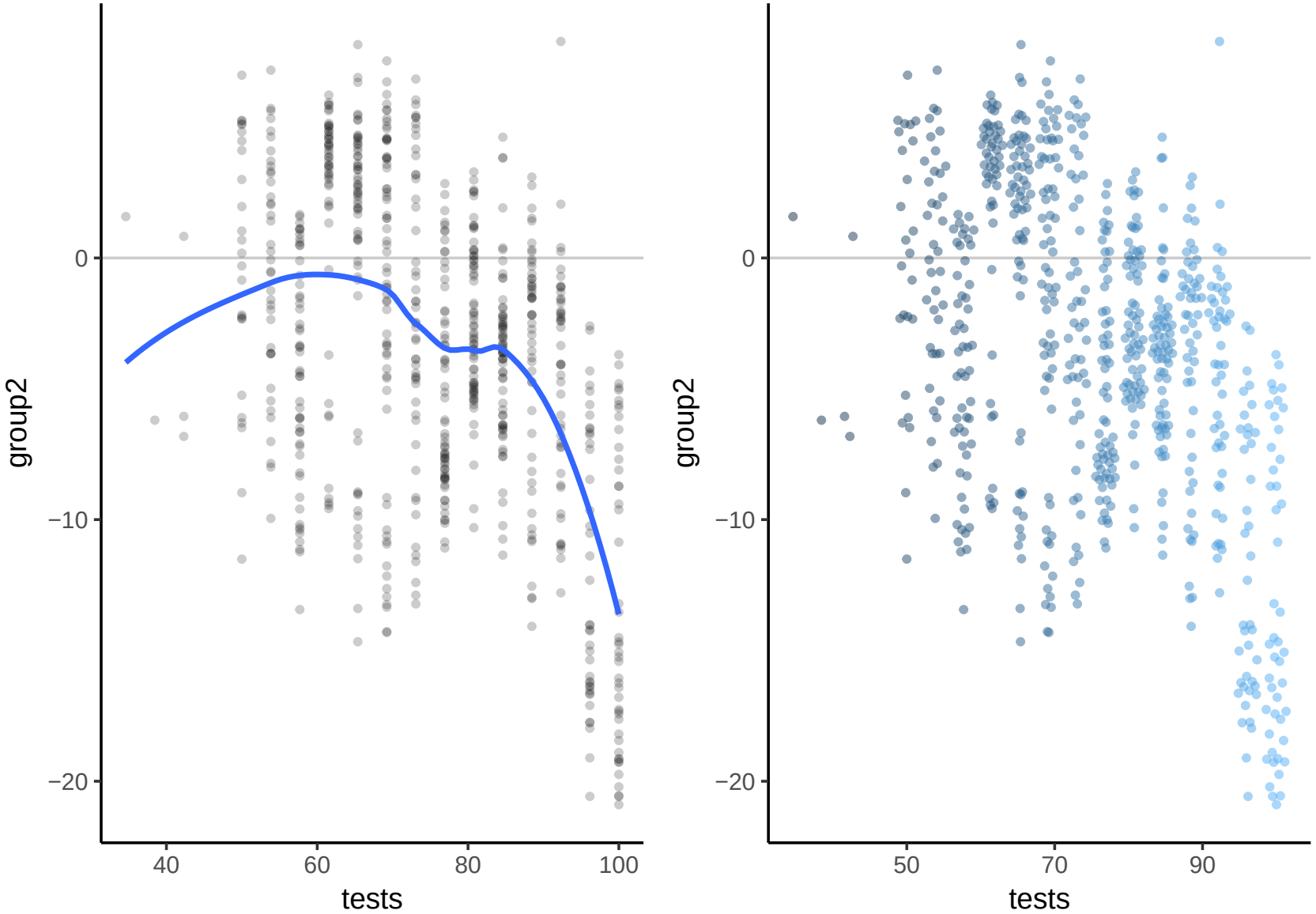
```
tree <- pmtree(base_model,
               control = ctree_control(maxdepth = 2))
plot(tree, terminal_panel = node_pmterminal(tree, confint = FALSE,
                                            plotfun = lmplot))
```



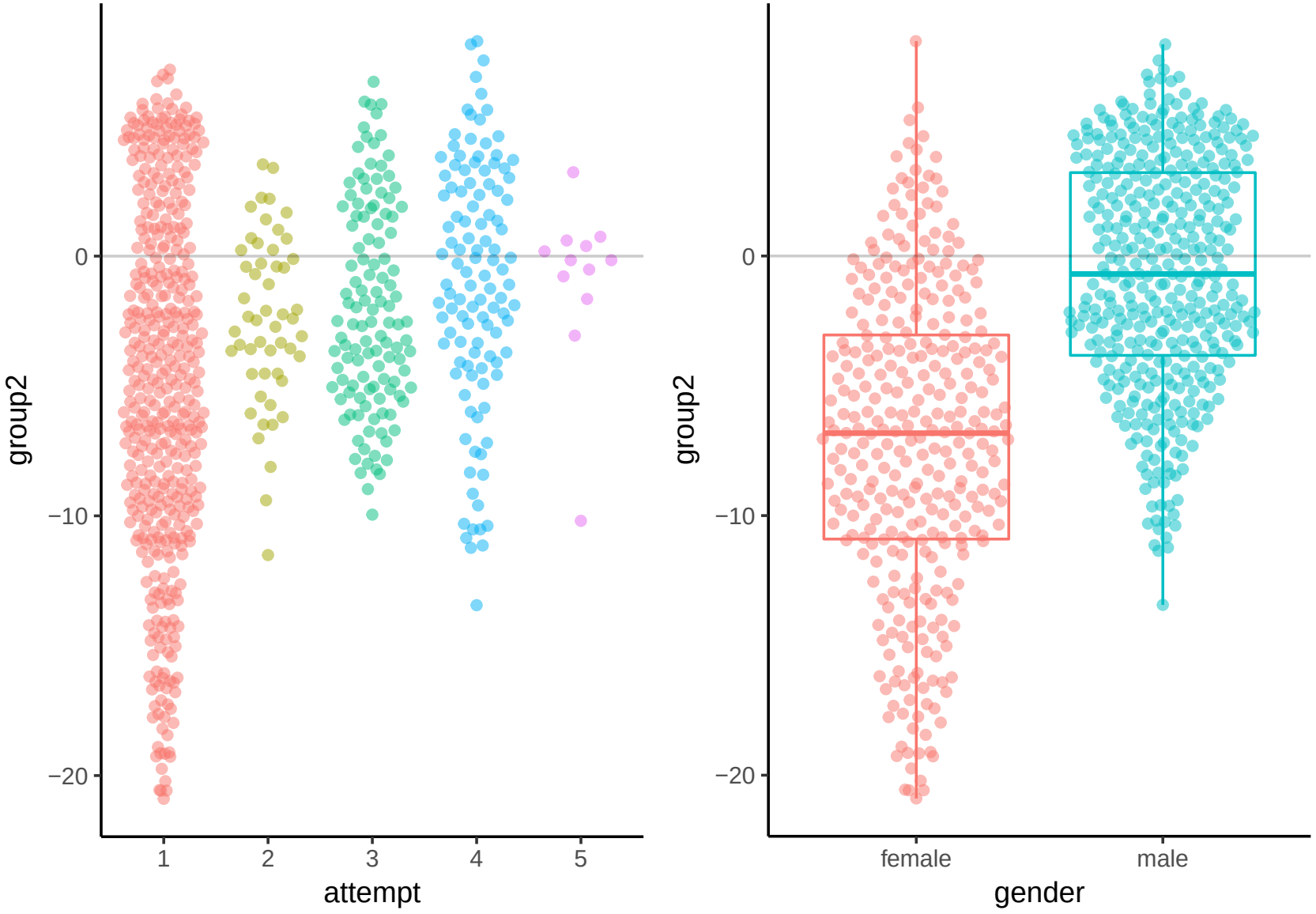
Math Exam: Personalised Treatment Effects

```
forest <- pmforest(base_model)
p_models <- pmodel(forest)

head(p_models)

##   (Intercept)     group2
## 1    54.50209 -10.393418
## 2    40.52339  -6.102260
## 3    54.25518  -8.054807
## 4    55.12411 -10.001973
## 5    62.76284  -6.374043
## 6    41.83793  -7.017029
```

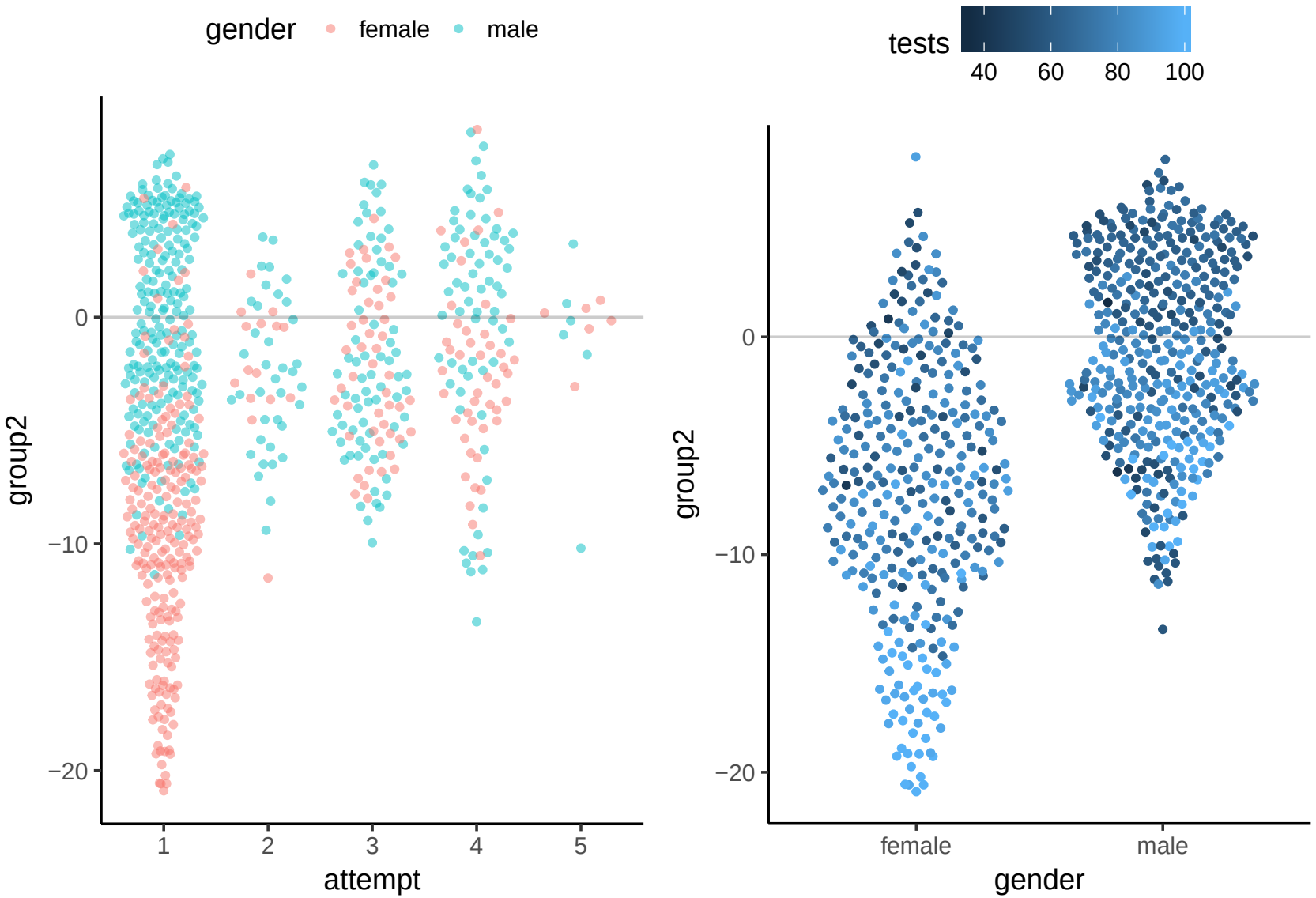
Math Exam: Personalised Treatment Effects



Math Exam: Personalised Treatment Effects



Math Exam: Personalised Treatment Effects



Related work

`partykit`	Model-based trees and forests in general
`trtf`	Transformation trees and forests
`disttree`	Distributional trees and forests
`glmertree`	Mixed model trees
`psychotree`	Psychometric model trees
`palmtree`	PALM (parially additive linear model) trees

Publications:

- H. Seibold, A. Zeileis, T. Hothorn
  Model-based Recursive Partitioning for Subgroup Analyses
  International Journal of Biostatistics, 2016.

- H. Seibold, A. Zeileis, T. Hothorn
  Individual Treatment Effect Prediction for ALS Patients
  Statistical Methods in Medical Research, 2017.

Data: `data("MathExam14W", package = "psychotools")`

Code: `https://r-forge.r-project.org/projects/partykit`

Contact: `heidi.seibold@uzh.ch`

The R Package `model4you`: Arguments

```
pmtree(object, data = NULL, zformula = ~.,
       control = ctree_control(), coeffun = coef, ...)
```

```
pmforest(object, data = NULL, zformula = ~., ntree = 100,
         control = ctree_control(lookahead = TRUE,
                                 mincriterion = 0,
                                 saveinfo = FALSE, ...),
         ...)

pmodel(x = NULL, object = NULL, newdata = NULL, OOB = TRUE,
       fun = coef, return_attr = c("modelcall", "data",
                                   "similarity"))
```