Biomarker analysis report

2017-08-29

Contents

1 The dataset

The dataset have 550 entries. In which 461 are in biomarker evaluable population (BEP).

- Endpoint of interest: Response
- Biomarker: KRAS.exprs
- Biomarker type: numeric

2 Representativeness: Selection Bias of Biomarker Population

In this section, we are trying to answer the question: *Are biomarker evaluable population representative of the full population population?*

Key baseline demographics and prognostic characteristics (including stratification variables and any variables with known prognostic effect) and efficacy outcomes should be summarized by treatment groups and compared between biomarker evaluable population (BEP) and the full population. These analyses are conducted to investigate any potential selection bias associated with the availability of the biomarker (e.g. we may not get enough tissue for patients whose tumor size is small. Therefore they may be exlcuded from BEP).

The key clinical biomarkers considered are:

```
clinical.vars.class
```

```
##           Sex           Age
## "categorical"     "numeric"
```

2.1 Check selection bias in terms of key clinical variables, between full population and BEP

```
kable(SummaryVars(data=input, trt=trt, subgroup=BEP, subgroup.indicator = BEP.indicator,
                  var=clinical.vars, var.class=clinical.vars.class))
```

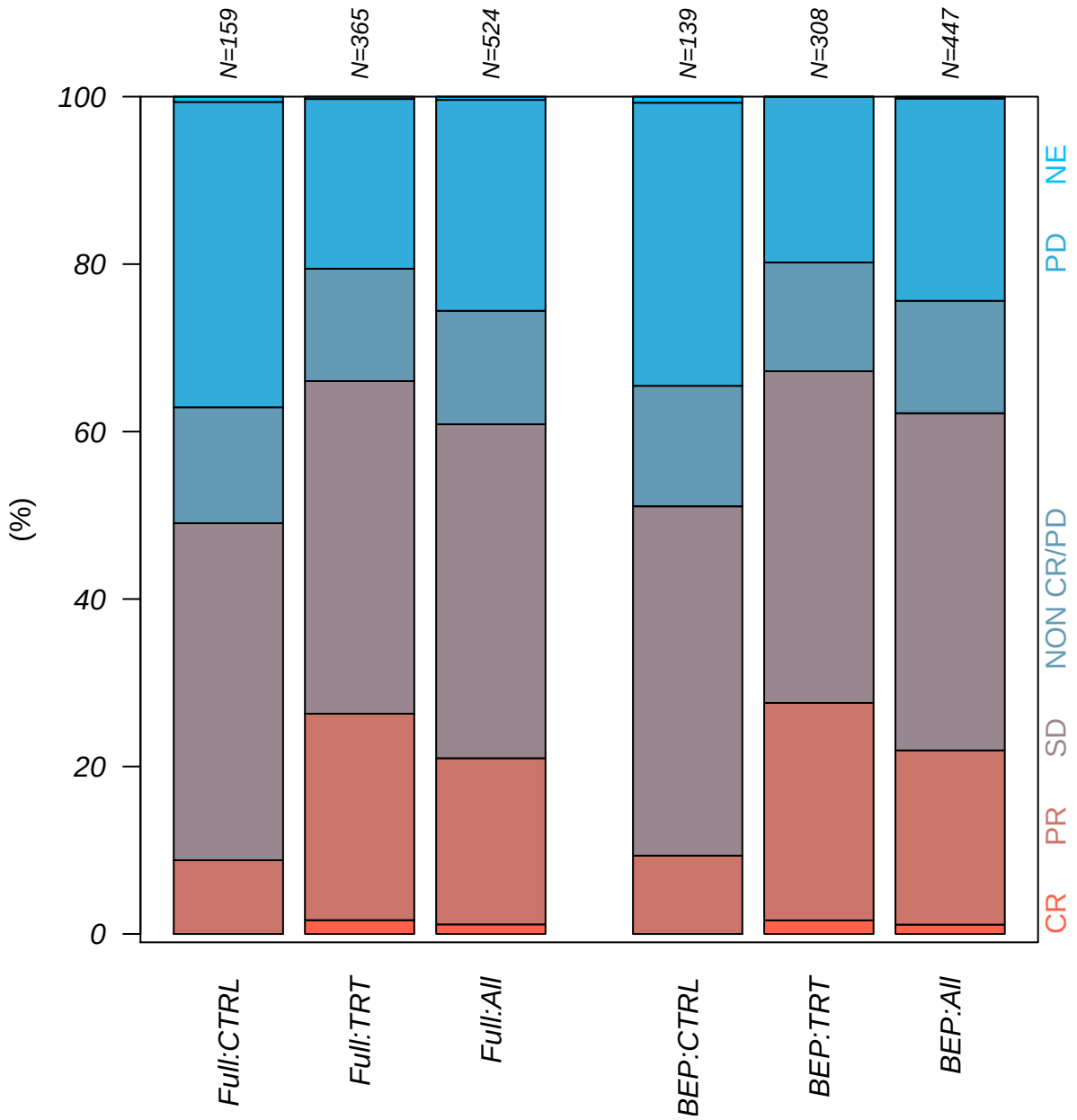
	All(CTRL)	BEP(CTRL)	All(TRT)	BEP(TRT)
Sex				
Total	182	152	368	309
NA's	0	0	0	0
F	89 (48.9%)	73 (48.03%)	184 (50%)	147 (47.57%)
M	93 (51.1%)	79 (51.97%)	184 (50%)	162 (52.43%)
Age				
N	182	152	368	309
Mean	52.54	52.35	54.03	54.29
Median	51.5	51	54	54
Min-Max	27...85	27...85	30...89	30...89
NA's	0	0	0	0

2.2 Check whether the clinical outcome in BEP is comparable to the full population

The following plot compares response outcome in BEP vs. the full population. The response category distribution is plotted for each arm. The BEP bars are expected to be comparable to those in full population.

```
Rsp.out <- PlotRspBar(data=input, outcome.var=outcome.var, binary=FALSE,
rsp.levels=rsp.levels, trt=trt, bep.indicator=BEP.indicator,
compare.bep.itt=TRUE, bep = BEP)
```

```
## entries with missing outcome.var are removed! 524 entries left
```




```
kable(Rsp.out$count,caption="count")
```

Table 2: count

	CR	PR	SD	NON CR/PD	PD	NE
Full:CTRL	0	14	64	22	58	1
Full:TRT	6	90	145	49	74	1
Full:All	6	104	209	71	132	2
BEP:CTRL	0	13	58	20	47	1
BEP:TRT	5	80	122	40	61	0
BEP:All	5	93	180	60	108	1

```
kable(round(Rsp.out$perc,2), caption="percentage")
```

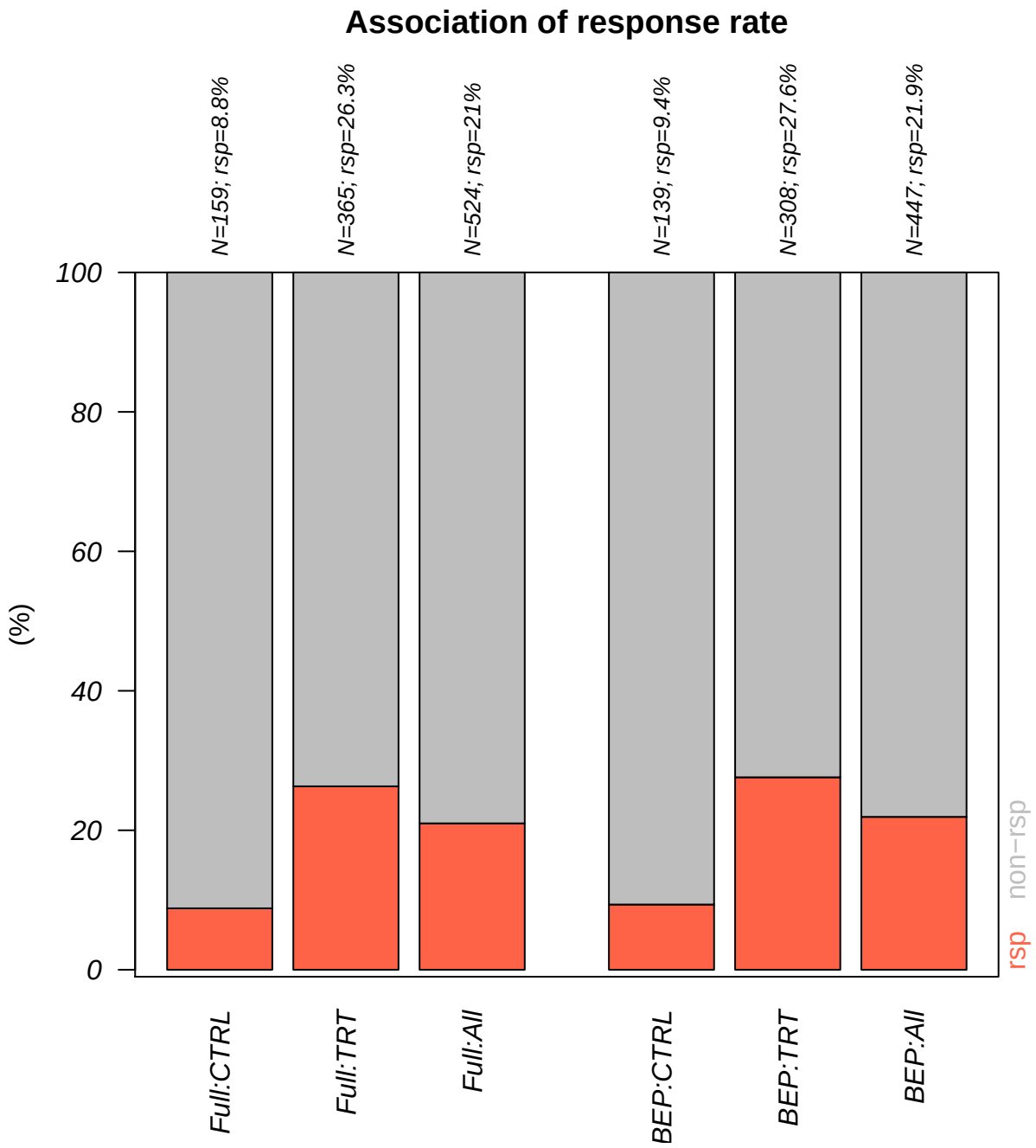
Table 3: percentage

	CR	PR	SD	NON CR/PD	PD	NE
Full:CTRL	0.00	0.09	0.40	0.14	0.36	0.01
Full:TRT	0.02	0.25	0.40	0.13	0.20	0.00
Full:All	0.01	0.20	0.40	0.14	0.25	0.00
BEP:CTRL	0.00	0.09	0.42	0.14	0.34	0.01
BEP:TRT	0.02	0.26	0.40	0.13	0.20	0.00
BEP:All	0.01	0.21	0.40	0.13	0.24	0.00

Summarize the response in two classes - rsp vs. non-rsp:

```
Rsp.out.2 <- PlotRspBar(data=input, outcome.var=outcome.var, binary=TRUE,
rsp.response = rsp.response,
rsp.nonresponse = rsp.nonresponse,rsp.name=rsp.name, trt=trt, bep.indicator=BEP.indicator,
compare.bep.itt=TRUE, bep = BEP)
```

```
## entries with missing outcome.var are removed! 524 entries left
```




```
kable(Rsp.out.2$count,caption="count")
```

Table 4: count

	rsp	non-rsp
Full:CTRL	14	145
Full:TRT	96	269
Full:All	110	414
BEP:CTRL	13	126
BEP:TRT	85	223
BEP:All	98	349

```
kable(round(Rsp.out.2$perc,2), caption="percentage")
```

Table 5: percentage

	rsp	non-rsp
Full:CTRL	0.09	0.91
Full:TRT	0.26	0.74
Full:All	0.21	0.79
BEP:CTRL	0.09	0.91
BEP:TRT	0.28	0.72
BEP:All	0.22	0.78

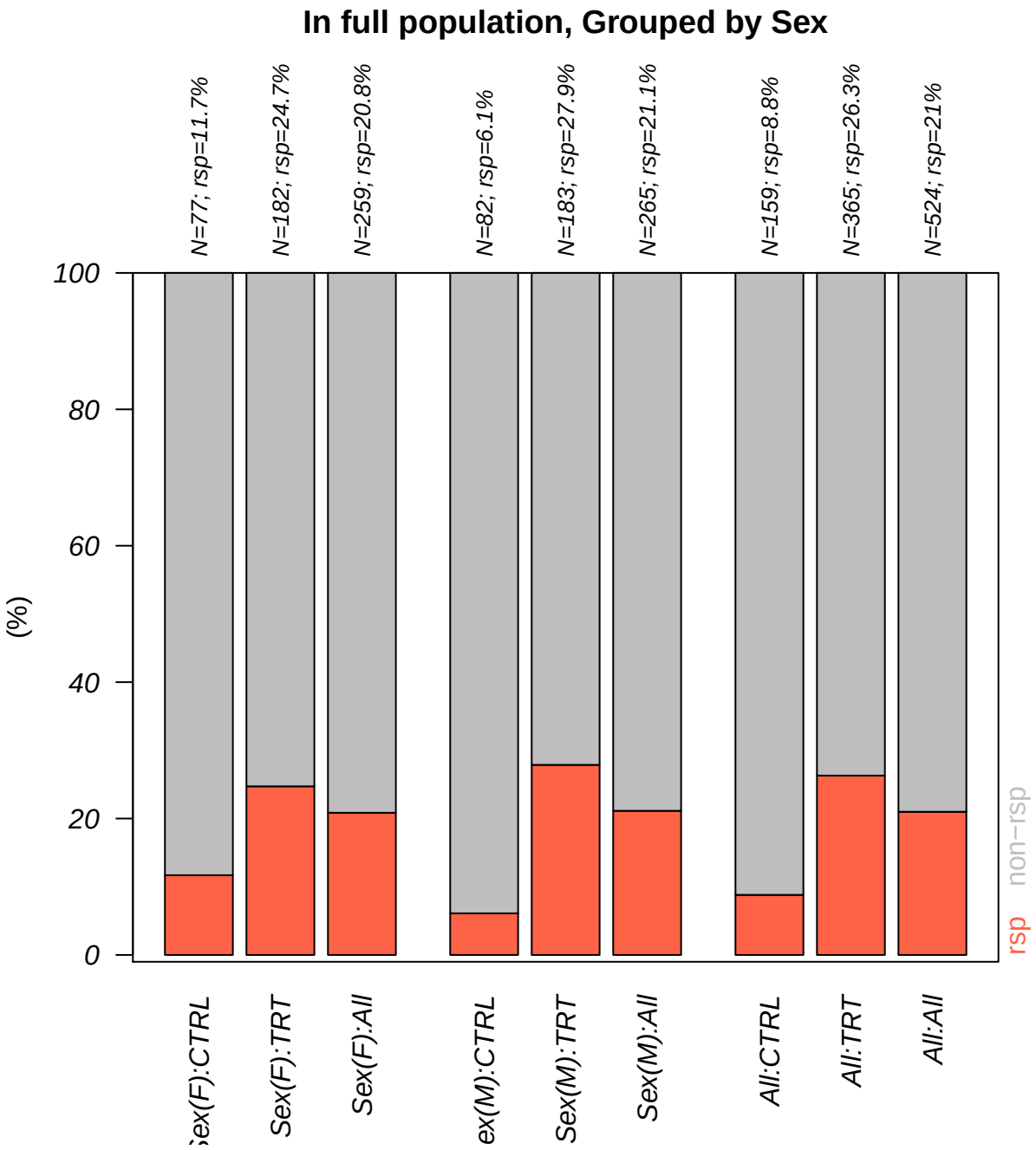
2.3 Examine whether the prognostic/predictive/null trend of key clinical variables holds in BEP

The following forest plot can be used to examine whether any of the key prognostic/predictive clinical variables still show prognostic/predictive trend in BEP:

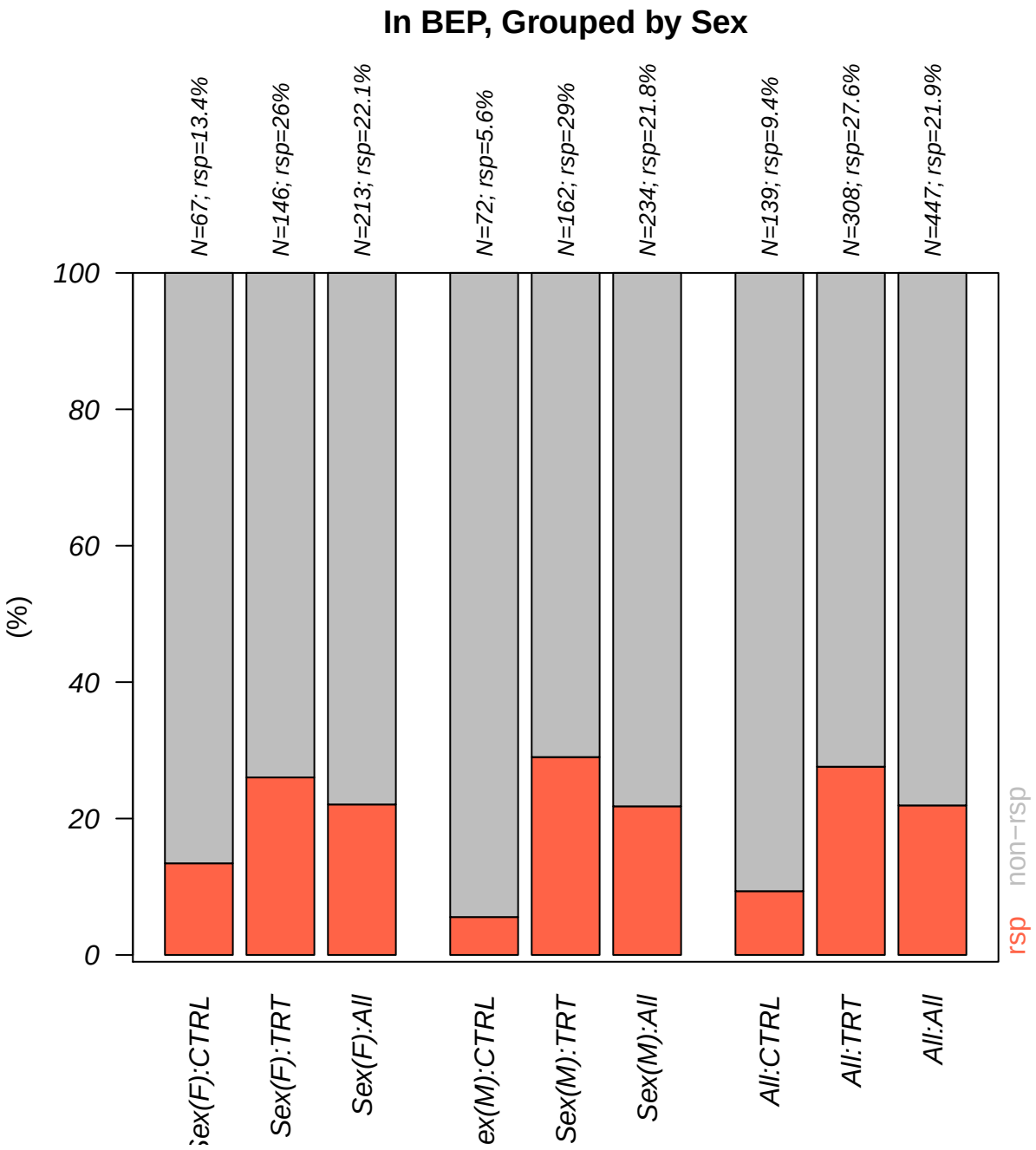
```
for(vv in clinical.vars.2){

  Rsp.out.tmp <- PlotRspBar(data=input, outcome.var=outcome.var, binary=TRUE,
  rsp.response = rsp.response,
  rsp.nonresponse = rsp.nonresponse,rsp.name=rsp.name, trt=trt,
  compare.var =TRUE,  var=vv, main=paste("In full population, Grouped by", vv))

  Rsp.out.tmp <- PlotRspBar(data=input.bep,
                            outcome.var=outcome.var, binary=TRUE,
  rsp.response = rsp.response,
  rsp.nonresponse = rsp.nonresponse,rsp.name=rsp.name, trt=trt,
  compare.var =TRUE, var=vv, main=paste("In BEP, Grouped by", vv))
}
```

```
## entries with missing outcome.var are removed! 524 entries left
```

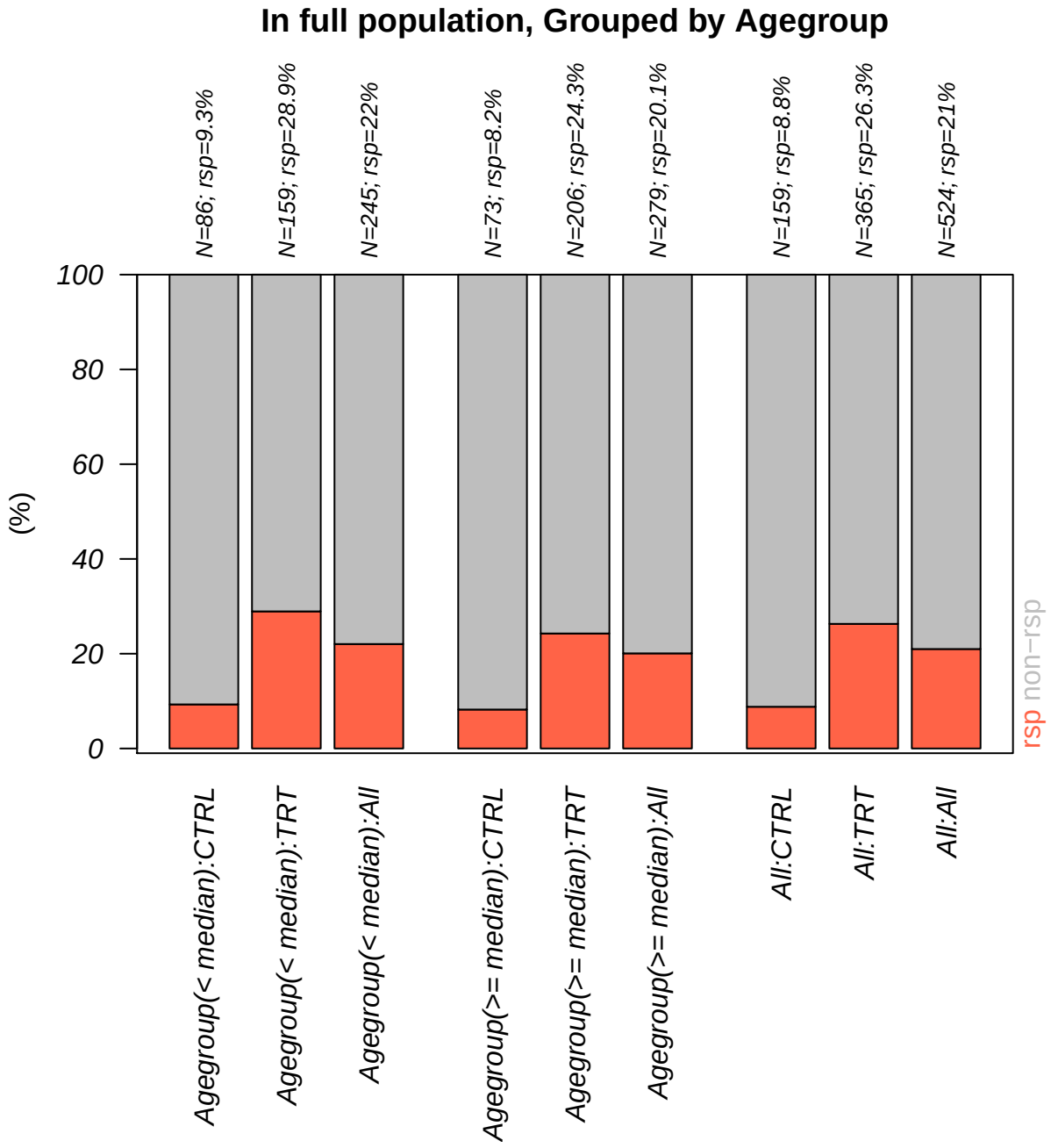
```
## entries with missing outcome.var are removed! 447 entries left
```

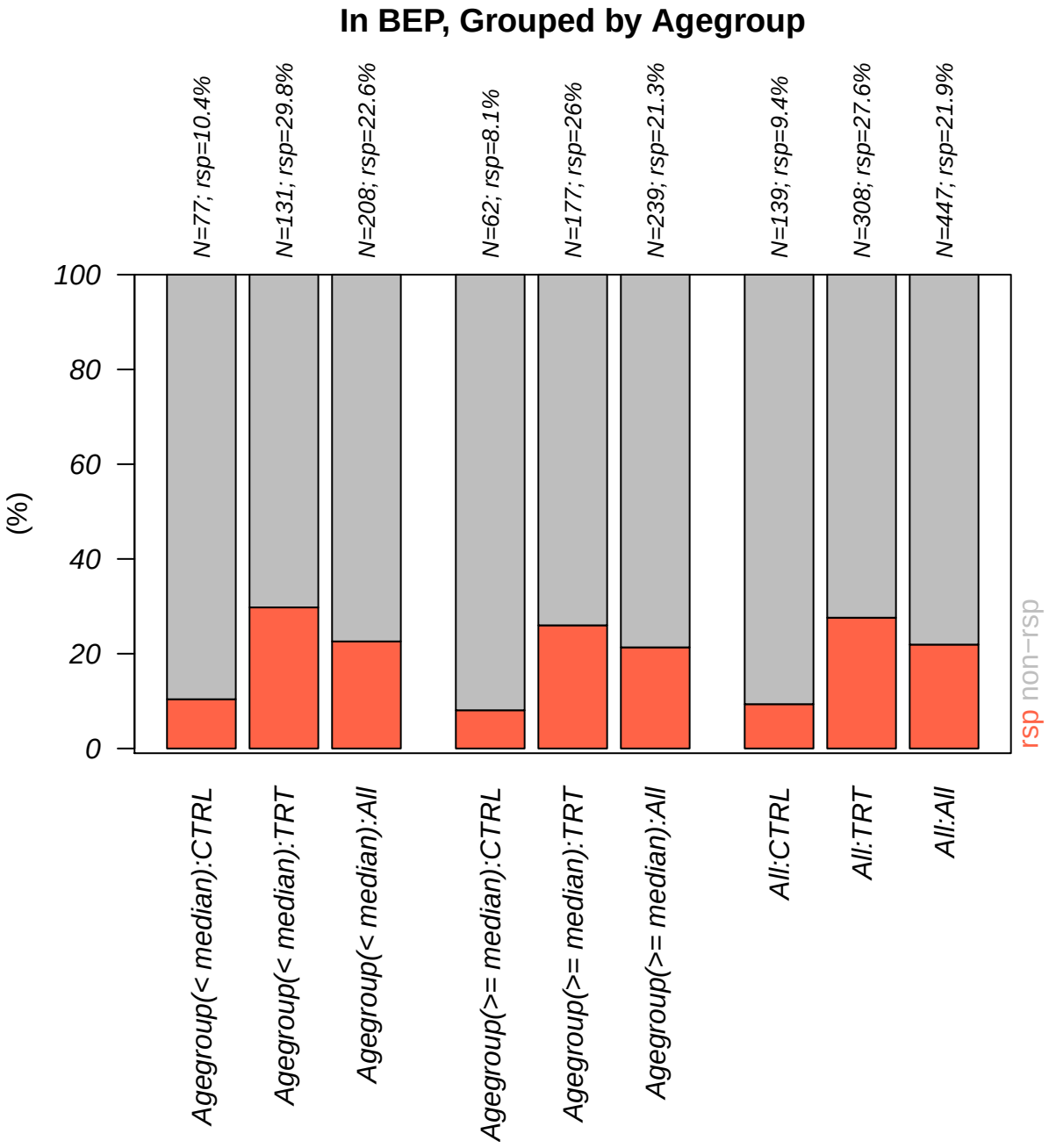



```
## entries with missing outcome.var are removed! 524 entries left
```




```
## entries with missing outcome.var are removed! 447 entries left
```


In full population, Grouped by Agegroup
N=86; rsp=9.3%
N=159; rsp=28.9%
N=245; rsp=22%
N=73; rsp=8.2%
N=206; rsp=24.3%
N=279; rsp=20.1%
N=159; rsp=8.8%
N=365; rsp=26.3%
N=524; rsp=21%
100
80
60
40
20
0
(%)
rsp
non−rsp
Agegroup(< median):CTRL
Agegroup(< median):TRT
Agegroup(< median):All
Agegroup(>= median):CTRL
Agegroup(>= median):TRT
Agegroup(>= median):All
All:CTRL
All:TRT
All:All



3 Biomarker property and its association to clinical variables

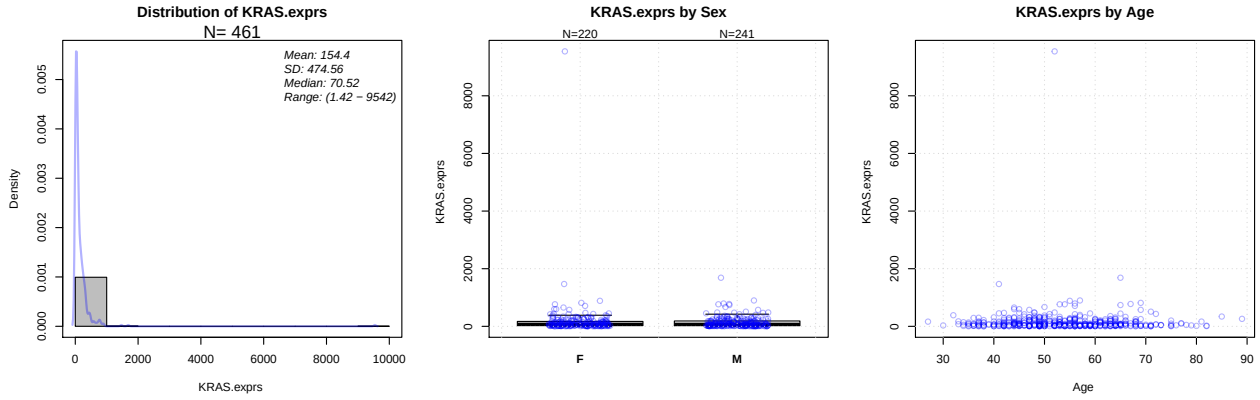
Before performing cutoff exploraotry analysis, it is important to check a biomarker's property. For example, whether this biomarker has a bi-modal or multi modal distribution - if so, this biomarker may has natural cutoff.

Relationship between the biomarker and key demographic and prognostic variables should also be investigated using bivariate plots. Prognostic property of the biomarker should also be assessed, by estimates of the clinical efficacy in the control arm.

3.1 Biomarker property and relationship to clinical variable

The following results show single variate plot for biomaker and bi-variate plots to investigate biomarker-clinical variable relationship.

```
PlotProperty(data=input, biomarker.var=bm, biomarker.class=bm.class,
             var=clinical.vars,
             var.class=clinical.vars.class,
             log2=FALSE, par.param = list(mfrow=c(2,3)))
```



3.2 Whether the biomarker shows within-arm effect

The following figures investigate whether the biomarker shows a within-arm effect (e.g. patients with higher biomarker value tend to have better clinical outcome):

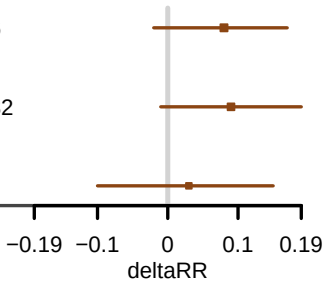
```
input.ctrl <- subset(input, Arm==placebo.code) ## Data with only ctrl samples
res.multicut.ctrl <- PlotTabForestBiomarker(data=input.ctrl,
                                outcome.class=outcome.class,
                                outcome.var=outcome.var,
                                var=bm,
                                var.class=bm.class,
                                percentile.cutoff=percentile.trycut,
                                numerical.cutoff = numerical.trycut,
                                rsp.response = rsp.response,
                                rsp.nonresponse = rsp.nonresponse,
                                main.prefix=placebo.code,
                                greater=TRUE, less=FALSE,
                                covariate=covariate, strata=strata)
```

```
## Covariate adjustment and stratification are not supported for binary outcome

## Some NAs in var column, will define the non NA entries as BEP

## only 1 arm; show.itt is set to FALSE

## only 1 arm; show.bep is set to FALSE

## Warning in prop.test(c(r1, r2), c(n1, n2), conf.level = 1 - alpha,
## correct = fit.para[["prop.test.use.continuity.correction"]]): Chi-squared
## approximation may be incorrect

## Warning in prop.test(c(r1, r2), c(n1, n2), conf.level = 1 - alpha,
## correct = fit.para[["prop.test.use.continuity.correction"]]): Chi-squared
```

```
## approximation may be incorrect
```

CTRL Within−arm Effect of Biomarker Response, KRAS.exprs Unadjusted, unstratified analysis

Subgroup	Group	nRsp/N	Rsp Rate	deltaRR	(CI)	raw P
KRAS.exprs(>=25%, 22.63)	Less	1 / 37	0.03			
	Greater	12 / 115	0.1	0.08	(−0.02 − 0.17)	0.26
KRAS.exprs(>=50%, 64)	Less	3 / 76	0.04			
	Greater	10 / 76	0.13	0.09	(−0.01 − 0.19)	0.082
KRAS.exprs(>=75%, 171.86)	Less	9 / 114	0.08			
	Greater	4 / 38	0.11	0.03	(−0.1 − 0.15)	0.87

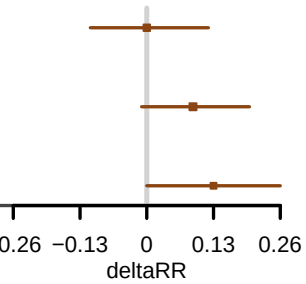



```
input.trt <- subset(input, Arm==active.code) ## Data with only ctrl samples
res.multicut.trt <- PlotTabForestBiomarker(data=input.trt,
                                    outcome.class=outcome.class,
                                    outcome.var=outcome.var,
                                    var=bm,
                                    var.class=bm.class,
                                    percentile.cutoff=percentile.trycut,
                                    numerical.cutoff = numerical.trycut,
                                    rsp.response = rsp.response,
                                    rsp.nonresponse = rsp.nonresponse,
                                    main.prefix=active.code,
                                    greater=TRUE, less=FALSE,
                                    covariate=covariate, strata=strata)
```

```
## Covariate adjustment and stratification are not supported for binary outcome
## Some NAs in var column, will define the non NA entries as BEP
## only 1 arm; show.itt is set to FALSE
## only 1 arm; show.bep is set to FALSE
## Warning in par(old.par): calling par(new=TRUE) with no plot
```

TRT Within−arm Effect of Biomarker Response, KRAS.exprs Unadjusted, unstratified analysis

Subgroup	Group	nRsp/N	Rsp Rate	deltaRR	(CI)	raw P
KRAS.exprs(>=25%, 25.28)	Less	21 / 77	0.27			
	Greater	64 / 232	0.28	0	(−0.11 − 0.12)	1
KRAS.exprs(>=50%, 72)	Less	35 / 153	0.23			
	Greater	50 / 156	0.32	0.09	(−0.01 − 0.2)	0.093
KRAS.exprs(>=75%, 182.28)	Less	56 / 231	0.24			
	Greater	29 / 78	0.37	0.13	(0 − 0.26)	0.039



The forest plots above show within-arm response rate difference (delta) between biomarker high (>= cutoff) vs. low (< cutoff) group. For a given arm, if the delta is not all around 0, it indicates that within this arm the biomarker has an association to the clinical outcome. For example, within treatment arm, suppose all high vs. low delta are greater than 0 and the delta is larger when cutting at a higher value. This indicates that among patients who received treatment, patients who have higher biomarker value tends to have better clinical outcome.

If similar trend is seen in both arms, it indicates that the biomarker may have a prognostic effect (the biomarker is able to identify patients with better/worse clinical outcome, regardless of treatment).

4 Biomarker cutoff exploration/selection

Results in this section could be used to examine multiple candidate cutoffs for a continuous biomarker. The need for cut-off determination should be rooted in the development strategy. In general, an exhaustive search looking at all possible cut-off values is not recommended for decision making. Over-optimized cutoff using one set of clinical data may lead to hard-to-reproduce results. When determining a cutoff, biomarker property should be considered - e.g. cut at a low-dense point may be more robust to population shift. The cutoff selection should also fit the program's stratigitic considerations. There is always a prevalence-effect size trade-off, inputs from multiple functions are needed - for example whether the team is willing to take more risk in PTS (high prevalence, weaker signal) or the team is willing to target at smaller population (lower prevalence, stronger signal)

4.1 Try different cutoffs - look for consistent trend

The following plots investigate whether the biomarker is predictive across arm. To perform cross-arm analysis. TRT-CTRL response rate delta within biomarker high or low group are calculated. The high/low groups are defined by trying different cutoffs.

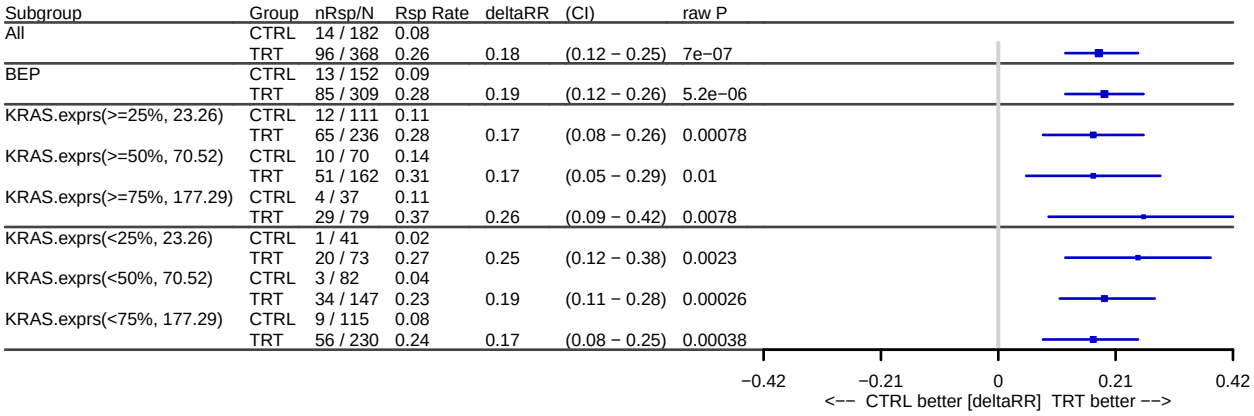
```
res.multicut <- PlotTabForestBiomarker(data=input,
                                       outcome.class=outcome.class,
                                       outcome.var=outcome.var,
                                       trt=trt,
                                       var=bm,
                                       var.class=bm.class,
                                       percentile.cutoff=percentile.trycut,
```

```
                                    numerical.cutoff = numerical.trycut,
                                    rsp.response = rsp.response,
                                    rsp.nonresponse = rsp.nonresponse,
                                    greater=TRUE, less=TRUE,
                                    show.itt=TRUE, show.bep=TRUE,
                                    covariate=covariate, strata=strata)
```

```
## Covariate adjustment and stratification are not supported for binary outcome
```

```
## Some NAs in var column, will define the non NA entries as BEP
```

Across−arm Effect of Biomarker Response, KRAS.exprs Unadjusted, unstratified analysis



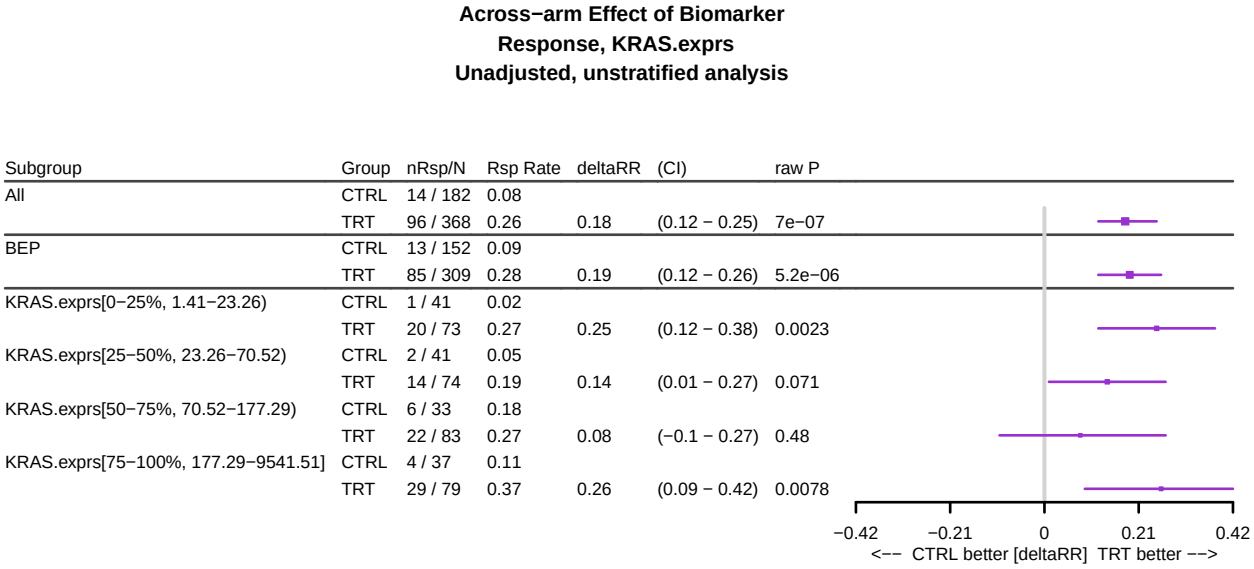
4.2 Estimations within non-overlapping bins

The following figure show TRT-CTRL response rate delta within non-overlapping bins defined by those exploratory cutoffs.

```
res.multicut <- PlotTabForestBiomarker(data=input,
                                    outcome.class=outcome.class,
                                    outcome.var=outcome.var,
                                    trt=trt,
                                    var=bm,
                                    var.class=bm.class,
                                    percentile.cutoff=percentile.trycut,
                                    numerical.cutoff = numerical.trycut,
                                    greater=FALSE, less=FALSE, within.bin=TRUE,
                                    show.itt=TRUE, show.bep=TRUE,
                                    covariate=covariate, strata=strata)
```

```
## Covariate adjustment and stratification are not supported for binary outcome
```

```
## Some NAs in var column, will define the non NA entries as BEP
```



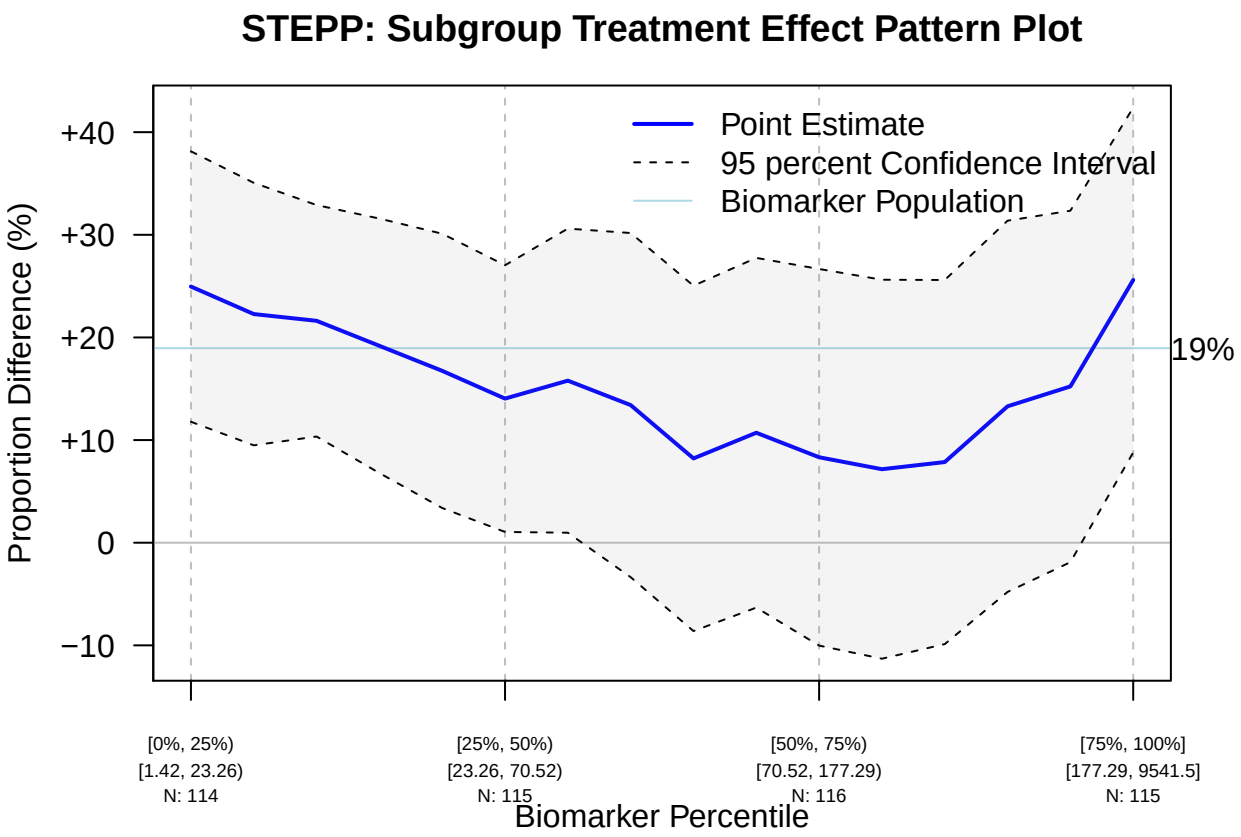
4.3 Estimations within overlapped sliding windows - STEPP plot

STEPP refers to Subgroup Treatment Effect Pattern Plot and it investigates relationship between biomarker and treatment effect. Only continuous biomarkers are suitable for STEPP analysis. STEPP performs treatment effect estimation on overlapping subsets of patients defined according to the biomarker level. The default setting of run.STEPP slides subgroup windows by 5% for each step and the subgroup size is 25% of the whole population.

A monotone trend is expected to be seen for an ideal biomarker.

```
stepp.out <- PlotSTEPP(data = input,
                                  outcome.class=outcome.class,
                                  outcome.var=outcome.var.bi,
                                  trt=trt,
                                  var=bm,
                                  placebo.code = placebo.code,
                                  active.code = active.code
)
```

```
## some NA in var column, will ignore NA entries
```



5 Biomarker subgroup analysis (using selected cutoff)

5.1 Estimations within each subgroup

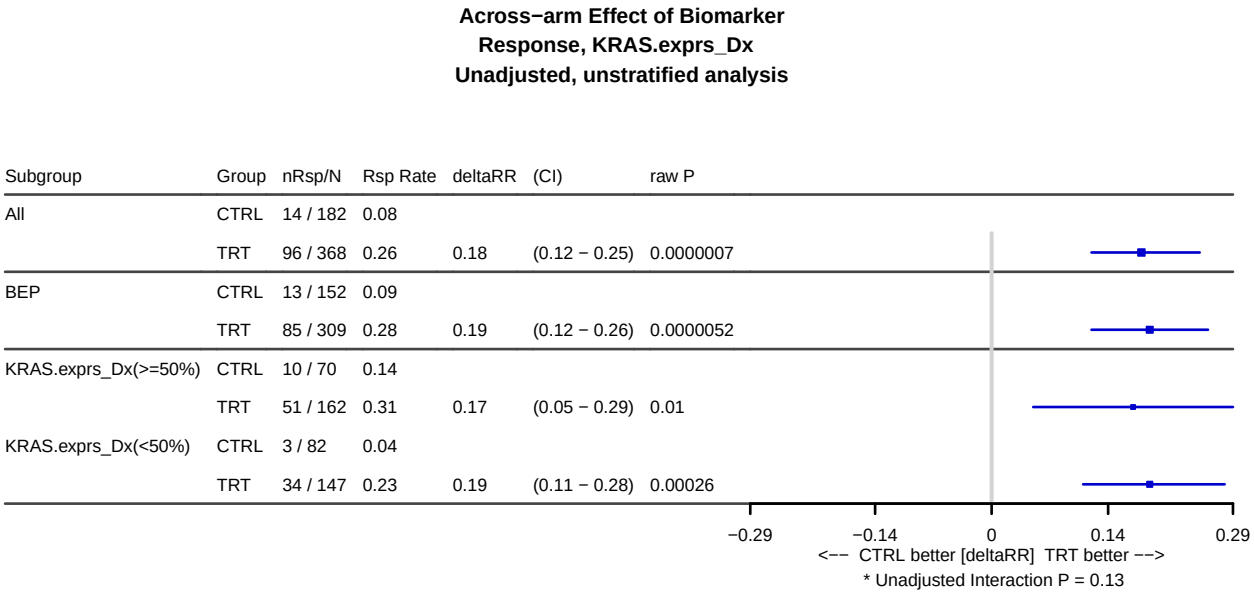
The following figure shows estimate of treatment effect in biomarker subgroups, defined by the selected cutoff.

```
if(bm.class=="numeric"){
if(!is.null(numerical.finalcut)) levs <- paste0(c(">=","<"),numerical.finalcut)
if(is.null(numerical.finalcut)) {
  nm <- quantile(input.bep[[bm]],percentile.finalcut, 2) # default quantile type in forest plot functio
  numerical.finalcut <- round(nm,2) # default rounding decimal in forest plots
  levs <- paste0(c(">=","<"),percentile.finalcut*100,"%")
}

bm2 <- paste0(bm,"_Dx")
input[[bm2]] <- ifelse(input[[bm]] >= numerical.finalcut, levs[1],levs[2])
input[[bm2]]<- factor(input[[bm2]], levels=levs) # ">=" as Dx+
}

if(bm.class=="categorical") {
  bm2 <- bm
  levs <- unique(input[[bm2]])
}

res.2group <- PlotTabForestBiomarker(data=input,
                                     outcome.class=outcome.class,
```

```
                                    outcome.var=outcome.var,
                                    trt=trt,
                                    var=bm2,
                                    var.class="categorical",
                                    greater=TRUE, less=TRUE,
                                    show.itt=TRUE, show.bep=TRUE,
                                    covariate=covariate, strata=strata)
```

```
## Covariate adjustment and stratification are not supported for binary outcome
```

```
## Some NAs in var column, will define the non NA entries as BEP
```

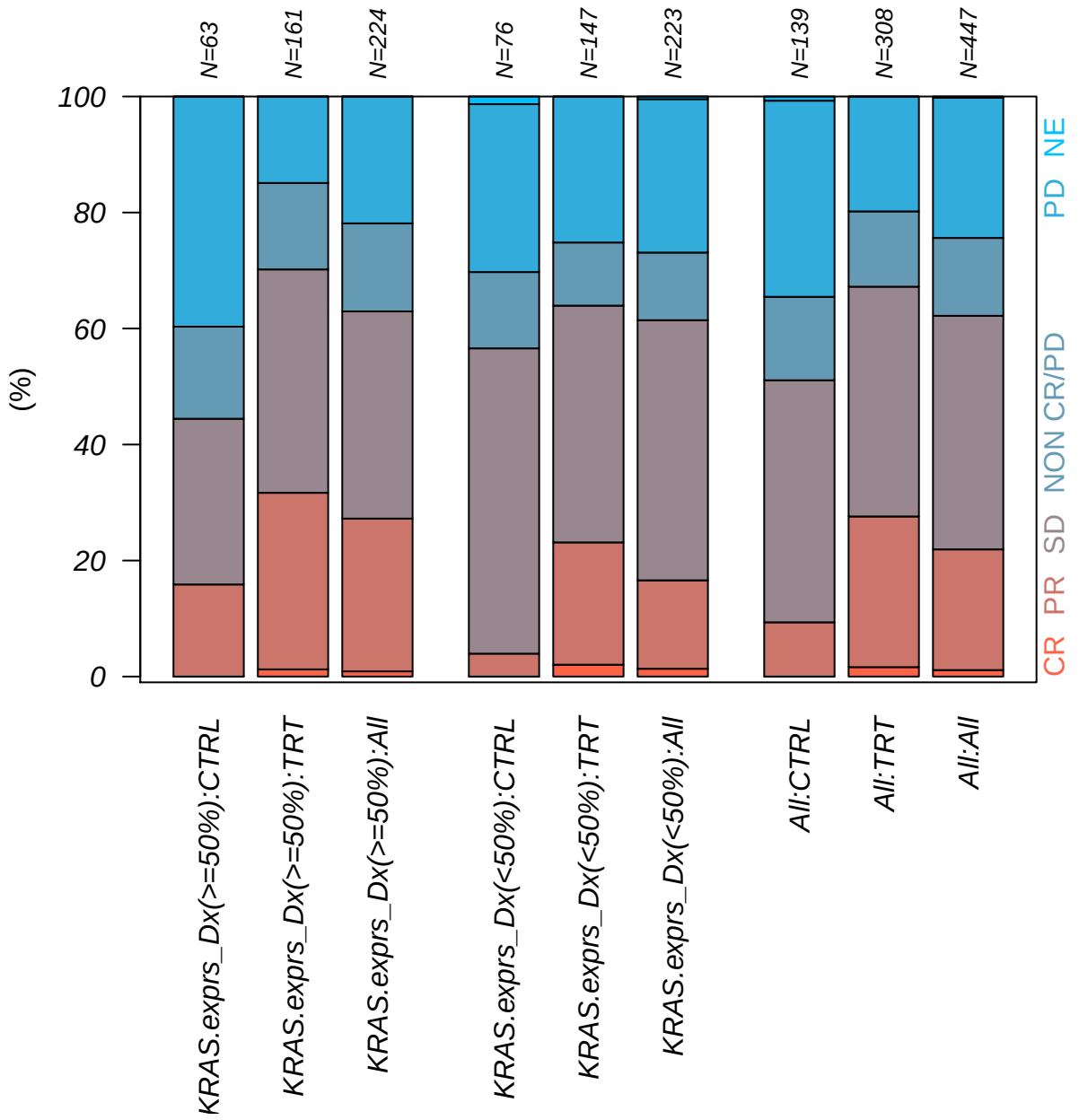


5.2 Subgroup analysis

The following figure show response category distributions of the biomarker subgroups, based on selected cutoff:

```
Rsp.out <- PlotRspBar(data=input, outcome.var=outcome.var, binary=FALSE,
rsp.levels=rsp.levels, trt=trt, bep.indicator=BEP.indicator, var=bm2,
compare.var=TRUE, bep = BEP)
```

```
## entries with missing outcome.var are removed! 524 entries left
```




```
kable(Rsp.out$count,caption="count")
```

Table 6: count

	CR	PR	SD	NON CR/PD	PD	NE
KRAS.exprs_Dx(>=50%):CTRL	0	10	18	10	25	0
KRAS.exprs_Dx(>=50%):TRT	2	49	62	24	24	0
KRAS.exprs_Dx(>=50%):All	2	59	80	34	49	0
KRAS.exprs_Dx(<50%):CTRL	0	3	40	10	22	1
KRAS.exprs_Dx(<50%):TRT	3	31	60	16	37	0
KRAS.exprs_Dx(<50%):All	3	34	100	26	59	1

	CR	PR	SD	NON CR/PD	PD	NE
All:CTRL	0	13	58	20	47	1
All:TRT	5	80	122	40	61	0
All:All	5	93	180	60	108	1

```
kable(round(Rsp.out$perc,2), caption="percentage")
```

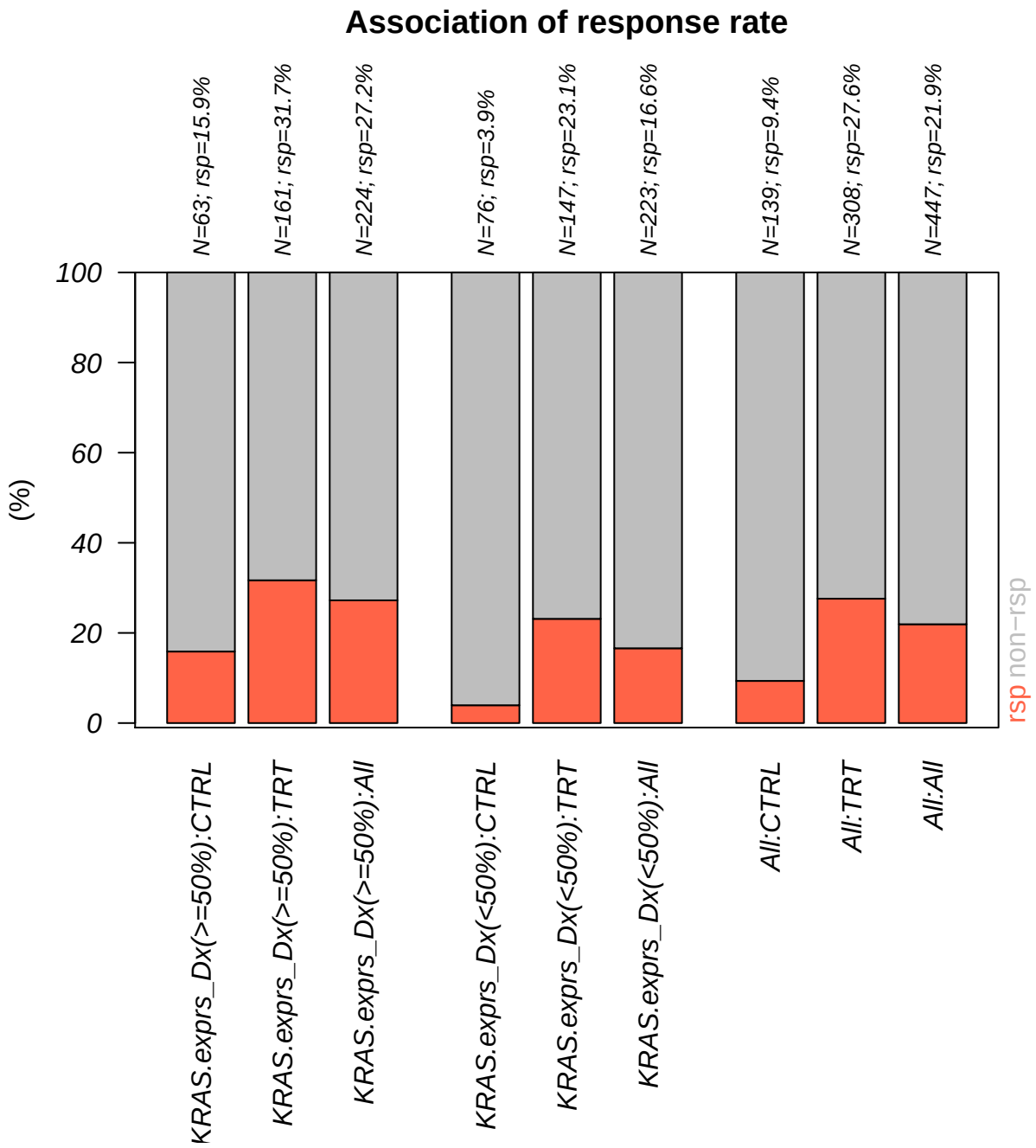
Table 7: percentage

	CR	PR	SD	NON CR/PD	PD	NE
KRAS.exprs_Dx(>=50%):CTRL	0.00	0.16	0.29	0.16	0.40	0.00
KRAS.exprs_Dx(>=50%):TRT	0.01	0.30	0.39	0.15	0.15	0.00
KRAS.exprs_Dx(>=50%):All	0.01	0.26	0.36	0.15	0.22	0.00
KRAS.exprs_Dx(<50%):CTRL	0.00	0.04	0.53	0.13	0.29	0.01
KRAS.exprs_Dx(<50%):TRT	0.02	0.21	0.41	0.11	0.25	0.00
KRAS.exprs_Dx(<50%):All	0.01	0.15	0.45	0.12	0.26	0.00
All:CTRL	0.00	0.09	0.42	0.14	0.34	0.01
All:TRT	0.02	0.26	0.40	0.13	0.20	0.00
All:All	0.01	0.21	0.40	0.13	0.24	0.00

Summarize the response in two classes - rsp vs. non-rsp:

```
Rsp.out.2 <- PlotRspBar(data=input, outcome.var=outcome.var, binary=TRUE,
rsp.response = rsp.response,
rsp.nonresponse = rsp.nonresponse,rsp.name=rsp.name, trt=trt, bep.indicator=BEP.indicator,
var=bm2, compare.var=TRUE, bep = BEP)
```

```
## entries with missing outcome.var are removed! 524 entries left
```




```
kable(Rsp.out.2$count,caption="count")
```

Table 8: count

	rsp	non-rsp
KRAS.exprs_Dx(>=50%):CTRL	10	53
KRAS.exprs_Dx(>=50%):TRT	51	110
KRAS.exprs_Dx(>=50%):All	61	163
KRAS.exprs_Dx(<50%):CTRL	3	73
KRAS.exprs_Dx(<50%):TRT	34	113
KRAS.exprs_Dx(<50%):All	37	186
All:CTRL	13	126

	rsp	non-rsp
All:TRT	85	223
All:All	98	349

```
kable(round(Rsp.out.2$perc,2), caption="percentage")
```

Table 9: percentage

	rsp	non-rsp
KRAS.exprs_Dx(>=50%):CTRL	0.16	0.84
KRAS.exprs_Dx(>=50%):TRT	0.32	0.68
KRAS.exprs_Dx(>=50%):All	0.27	0.73
KRAS.exprs_Dx(<50%):CTRL	0.04	0.96
KRAS.exprs_Dx(<50%):TRT	0.23	0.77
KRAS.exprs_Dx(<50%):All	0.17	0.83
All:CTRL	0.09	0.91
All:TRT	0.28	0.72
All:All	0.22	0.78

5.3 Check whether biomarker subgroup is confounded with key clinical variables

The following table checks whether the biomarker subgroup is confounded with clinical variables. Here we check whether the clinical variable distribution is comparable in biomarker high and low group.

```
input.bep <- input[which(input[[BEP]]==BEP.indicator),]


kable(
SummaryVars(data=input.bep,trt=trt, subgroup=bm2, var=clinical.vars,
       var.class=clinical.vars.class, subgroup.indicator=levs[1],compare.subgroup=TRUE)
)
```

	KRAS.exprs_Dx_>=50%(CTRL)	KRAS.exprs_Dx_<50%(CTRL)	KRAS.exprs_Dx_>=50%(TRT)	KRAS.exprs_Dx_<50%(TRT)
Sex				
Total	70	82	162	147
NA's	0	0	0	0
F	35 (50%)	38 (46.34%)	77 (47.53%)	70 (47.62%)
M	35 (50%)	44 (53.66%)	85 (52.47%)	77 (52.38%)
Age				
N	70	82	162	147
Mean	51.5	53.07	53.36	55.31
Median	50	52	53	55
Min-Max	27...85	35...82	33...89	30...82
NA's	0	0	0	0