Messina E2B: MessinaSurv vs Others

March 30, 2015

1 Preparation

```
library(plyr)
library(messina)
library(maxstat)
library(doMC)

deltaForMargin = function(margin, sigma_epsilon = 1, alpha = 0.05) margin - 2*sigma_epsilon*qnorm(alpha)
marginForDelta = function(delta, sigma_epsilon = 1, alpha = 0.05) delta + 2*sigma_epsilon*qnorm(alpha)

messina_objectives = list("cox.log2" = messinaSurvObj.CoxCoef(log(2)))

e2b.design = expand.grid(
        Delta = seq(0, 5, 0.5),
        R1 = c(1, 2, 4, 8, 16),
        sigma_epsilon = 1,
        p1 = c(0.2, 0.5, 0.8),
        pc = c(0, 0.2, 0.5),
        alpha = 0.2,
        stat.alpha = 0.05,
        messina.objective = "cox.log2",
        messina.minmarg = 1,
        n = c(25, 50, 100),
        reps = 5e1)

e2b.design$margin = marginForDelta(e2b.design$Delta, e2b.design$sigma_epsilon, alpha = e2b.design$alpha)

detector_multicut = function(x, y, ncuts = 10, correct = "none")
{
        if (ncuts == 1)         { correct = "none" }
        if (is.vector(x))       { x = matrix(x, nrow = 1) }

        aaply(x, 1, function(x1) {
                cutpoints = quantile(x1, probs = (1:ncuts)/(ncuts + 1))
                pvals = sapply(cutpoints, function(c) {
                        x1c = x1 > c
                        test = survdiff(y ~ x1c)
                        pval = pchisq(test$chisq, df = 1, lower.tail = FALSE)
                        pval
                })
                pvals = p.adjust(pvals, correct)
                pvals[is.na(pvals)] = 1
```

```
                min(pvals)
        }, .parallel = FALSE)
}

e2b.datafun = function(n, p1, pc, Delta, R1, sigma_epsilon, ...)
{
        n1 = round(n*p1)
        n0 = n - n1
        c = rep(c(0, 1), c(n0, n1))
        x = Delta*c + rnorm(n, mean = 0, sd = sigma_epsilon)

        Rc = optimize(function(Rc) abs(integrate(function(t) pexp(t, Rc) * ((1-p1)*dexp(t, 1) + p1*dexp(

        time_event = c(rexp(n0, 1), rexp(n1, R1))
        time_cens = rexp(n, Rc)

        y = Surv(pmin(time_event, time_cens), time_event <= time_cens)

        list(x = x, y = y, c = c)
}

e2b.detfun = function(x, y, stat.alpha, messina.objective, messina.minmarg, ...)
{
        data.maxstat = data.frame(time = y[,1], event = y[,2], x = x)
        test.maxstat = try(maxstat.test(Surv(time, event) ~ x, data = data.maxstat, smethod = "LogRank",
        det.maxstat = ifelse(class(test.maxstat) == "try-error", FALSE, test.maxstat$p.value < stat.alph

        det.1cut = detector_multicut(x, y, 1) < stat.alpha
        det.10cut = detector_multicut(x, y, 10) < stat.alpha
        det.10cutHolm = detector_multicut(x, y, 10, "holm") < stat.alpha

        x.messina = rbind(x, x, x)
        fit.messina = messinaSurv(x.messina, y, messina_objectives[[messina.objective]], silent = TRUE,
        det.messina = fit.messina@fits@summary$passed[1] == TRUE && fit.messina@fits@summary$margin[1] >
        det.messina.nomarg = fit.messina@fits@summary$passed[1] == TRUE

        c(maxstat = det.maxstat, c1 = det.1cut, c10 = det.10cut, cH10 = det.10cutHolm, m = det.messina,
}

e2b.repfun = function(i, Delta, R1, sigma_epsilon, pc, p1, stat.alpha, messina.minmarg, messina.objectiv
{
        data = e2b.datafun(n, p1, pc, Delta, R1, sigma_epsilon)
        dets = try(e2b.detfun(data$x, data$y, stat.alpha, messina.objective, messina.minmarg))
        if(class(dets) == "try-error")  { return(c(NA, NA, NA, NA, NA, NA)) }
        return(dets)
}

e2b.expfun = function(Delta, R1, sigma_epsilon, pc, p1, stat.alpha, messina.minmarg, messina.objective,
{
        message(date(), "\t", jobindex)
        detections = sapply(1:reps, e2b.repfun, Delta = Delta, R1 = R1, sigma_epsilon = sigma_epsilon, p
        detrate = rowMeans(detections, na.rm = TRUE)
        detrate
```

```
}

registerDoMC(32)

e2b.design$jobindex = 1:nrow(e2b.design)
set.seed(20150320)
e2b.det = mlply(e2b.design, e2b.expfun, .parallel = TRUE)

e2b.design = rbind(
        cbind(e2b.design, method = "maxstat", detrate = simplify2array(e2b.det)[1,]),
        cbind(e2b.design, method = "c1", detrate = simplify2array(e2b.det)[2,]),
        cbind(e2b.design, method = "c10", detrate = simplify2array(e2b.det)[3,]),
        cbind(e2b.design, method = "cH10", detrate = simplify2array(e2b.det)[4,]),
        cbind(e2b.design, method = "messina", detrate = simplify2array(e2b.det)[5,]),
        cbind(e2b.design, method = "messina_0m", detrate = simplify2array(e2b.det)[6,])
)
```

```
e2b.design2 = expand.grid(
        Delta = seq(0, 5, 0.5),
        R1 = c(1, 4, 16),
        sigma_epsilon = 1,
        p1 = c(0.2, 0.5),
        pc = c(0.2, 0.5),
        alpha = 0.2,
        stat.alpha = 0.05,
        messina.objective = "cox.log2",
        messina.minmarg = 1,
        n = c(25, 50, 100),
        reps = 5e2)

e2b.design2$margin = marginForDelta(e2b.design2$Delta, e2b.design2$sigma_epsilon, alpha = e2b.design2$al

registerDoMC(32)
e2b.design2$jobindex = 1:nrow(e2b.design2)
set.seed(20150321)
e2b.det2 = mlply(e2b.design2, e2b.expfun, .parallel = TRUE)

e2b.design2 = rbind(
        cbind(e2b.design2, method = "maxstat", detrate = simplify2array(e2b.det2)[1,]),
        cbind(e2b.design2, method = "c1", detrate = simplify2array(e2b.det2)[2,]),
        cbind(e2b.design2, method = "c10", detrate = simplify2array(e2b.det2)[3,]),
        cbind(e2b.design2, method = "cH10", detrate = simplify2array(e2b.det2)[4,]),
        cbind(e2b.design2, method = "messina", detrate = simplify2array(e2b.det2)[5,]),
        cbind(e2b.design2, method = "messina_0m", detrate = simplify2array(e2b.det2)[6,])
)
```

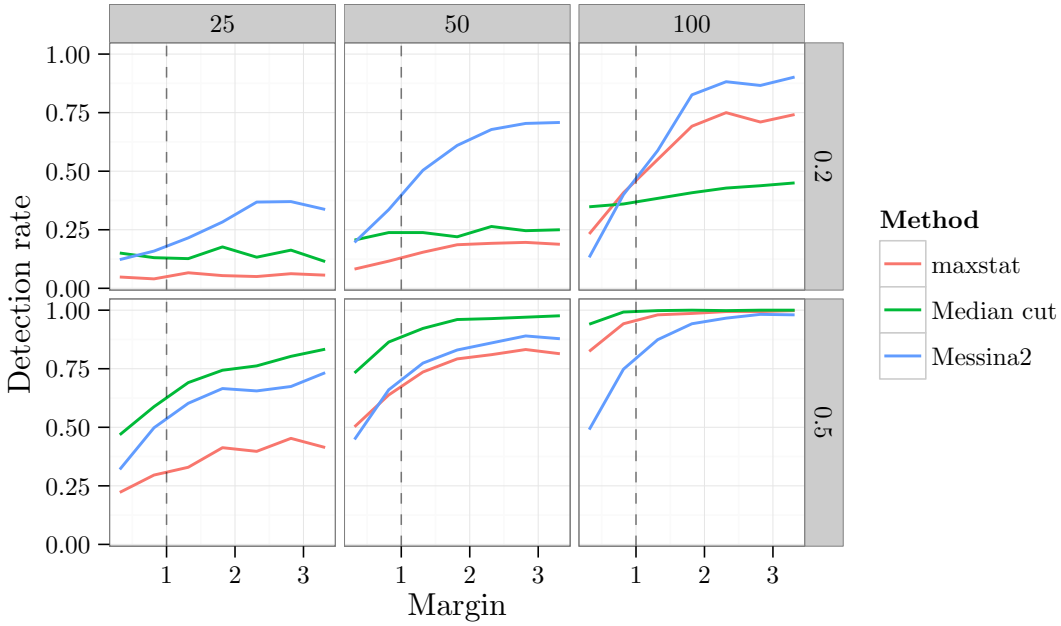
```
e2b.design3 = expand.grid(
        Delta = 100,
        R1 = 4,
        sigma_epsilon = 1,
        p1 = seq(0, 1, 0.05),
        pc = 0.2,
```

```
        alpha = 0.2,
        stat.alpha = 0.05,
        messina.objective = "cox.log2",
        messina.minmarg = 1,
        n = 50,
        reps = 5e3)

registerDoMC(32)
e2b.design3$jobindex = 1:nrow(e2b.design3)
set.seed(20150321)
e2b.det3 = mlply(e2b.design3, e2b.expfun, .parallel = TRUE)

e2b.design3 = rbind(
        cbind(e2b.design3, method = "maxstat", detrate = simplify2array(e2b.det3)[1,]),
        cbind(e2b.design3, method = "c1", detrate = simplify2array(e2b.det3)[2,]),
        cbind(e2b.design3, method = "c10", detrate = simplify2array(e2b.det3)[3,]),
        cbind(e2b.design3, method = "cH10", detrate = simplify2array(e2b.det3)[4,]),
        cbind(e2b.design3, method = "messina", detrate = simplify2array(e2b.det3)[5,]),
        cbind(e2b.design3, method = "messina_0m", detrate = simplify2array(e2b.det3)[6,])
)
```

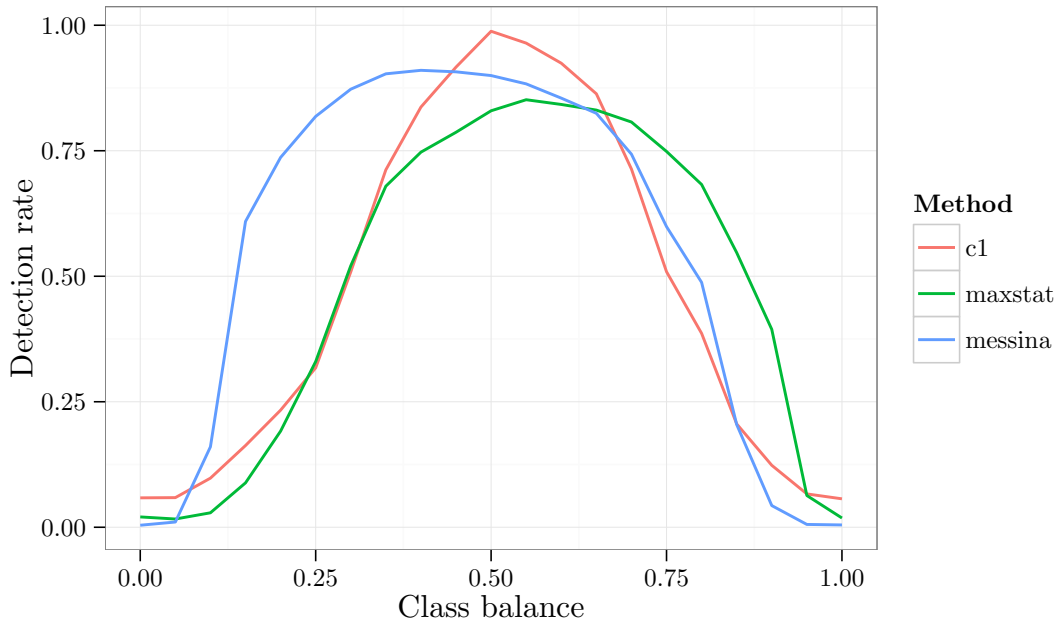
```
e2b.design4 = expand.grid(
        Delta = 100,
        R1 = 1,
        sigma_epsilon = 1,
        p1 = 0.5,
        pc = 0.2,
        alpha = 0.2,
        stat.alpha = 0.05,
        multicut.n = 1:25,
        n = 50,
        reps = 5e3)

e2b.detfun4 = function(x, y, multicut.n, stat.alpha, ...)
{
        det.multicut = detector_multicut(x, y, multicut.n) < stat.alpha
        det.multicutHolm = detector_multicut(x, y, multicut.n, "holm") < stat.alpha

        c(mc = det.multicut, mcH = det.multicutHolm)
}

e2b.repfun4 = function(i, Delta, R1, sigma_epsilon, pc, p1, multicut.n, stat.alpha, n, ...)
{
        data = e2b.datafun(n, p1, pc, Delta, R1, sigma_epsilon)
        dets = try(e2b.detfun4(data$x, data$y, multicut.n, stat.alpha))
        if(class(dets) == "try-error")  { return(c(NA, NA)) }
        return(dets)
}

e2b.expfun4 = function(Delta, R1, sigma_epsilon, pc, p1, multicut.n, stat.alpha, n, reps, ...)
{
        detections = sapply(1:reps, e2b.repfun4, Delta = Delta, R1 = R1, sigma_epsilon = sigma_epsilon,
        detrate = rowMeans(detections, na.rm = TRUE)
```

```
        detrate
}

registerDoMC(32)
set.seed(20150321)
e2b.det4 = mlply(e2b.design4, e2b.expfun4, .parallel = TRUE)

e2b.design4 = rbind(
        cbind(e2b.design4, method = "mc", detrate = simplify2array(e2b.det4)[1,]),
        cbind(e2b.design4, method = "mcH", detrate = simplify2array(e2b.det4)[2,])
)
```

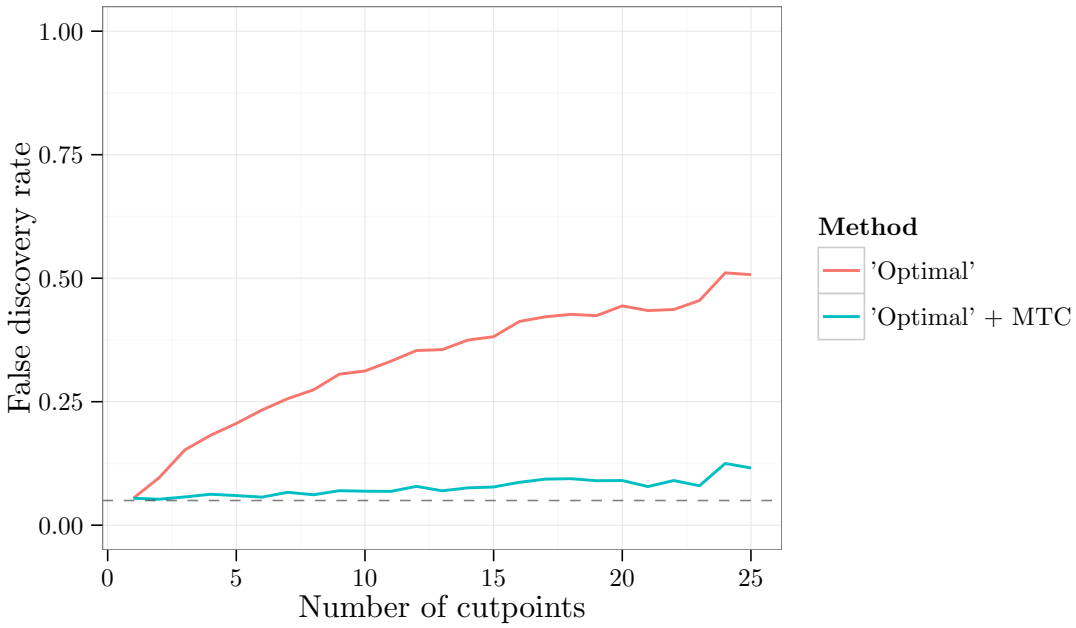
```
library(ggplot2)

e2b.design$method = as.character(e2b.design$method)
e2b.design$method[e2b.design$method == "messina_0m"] = "messina0m"
e2b.design$method = as.factor(e2b.design$method)
e2b.design2$method = as.character(e2b.design2$method)
e2b.design2$method[e2b.design2$method == "messina_0m"] = "messina0m"
e2b.design2$method = as.factor(e2b.design2$method)
e2b.design3$method = as.character(e2b.design3$method)
e2b.design3$method[e2b.design3$method == "messina_0m"] = "messina0m"
e2b.design3$method = as.factor(e2b.design3$method)
e2b.design4$method = as.character(e2b.design4$method)
e2b.design4$method[e2b.design4$method == "mc"] = "'Optimal'"
e2b.design4$method[e2b.design4$method == "mcH"] = "'Optimal' + MTC"
e2b.design4$method = as.factor(e2b.design4$method)

# ggplot(e2b.design[e2b.design£n == 25,], aes(x = margin, y = detrate, colour = factor(method))) + geom_
# ggplot(e2b.design[e2b.design£n == 50,], aes(x = margin, y = detrate, colour = factor(method))) + geom_
# ggplot(e2b.design[e2b.design£n == 100,], aes(x = margin, y = detrate, colour = factor(method))) + geo

# ggplot(e2b.design2[e2b.design2£n == 25,], aes(x = margin, y = detrate, colour = factor(method))) + ge
# ggplot(e2b.design2[e2b.design2£n == 50,], aes(x = margin, y = detrate, colour = factor(method))) + ge
# ggplot(e2b.design2[e2b.design2£n == 100,], aes(x = margin, y = detrate, colour = factor(method))) + g

e2b.design2$method = as.character(e2b.design2$method)
e2b.design2$method[e2b.design2$method == "c1"] = "Median cut"
e2b.design2$method[e2b.design2$method == "messina"] = "Messina2"
ggplot(e2b.design2[e2b.design2$margin >= 0 & e2b.design2$R1 == 4 & e2b.design2$pc == 0.2 & e2b.design2$m
```




```
ggplot(e2b.design3[e2b.design3$method %in% c("c1", "maxstat", "messina"),], aes(x = p1, y = detrate, col
```




```
ggplot(e2b.design4, aes(x = multicut.n, y = detrate, colour = factor(method))) + geom_line(lwd = 1) + xl
```


1.00
0.75
0.50
0.25
0.00
False discovery rate
0
5
10
15
20
25
Number of cutpoints
Method
'Optimal'
'Optimal' + MTC