

pCR Meta Analysis

Michael Sachs

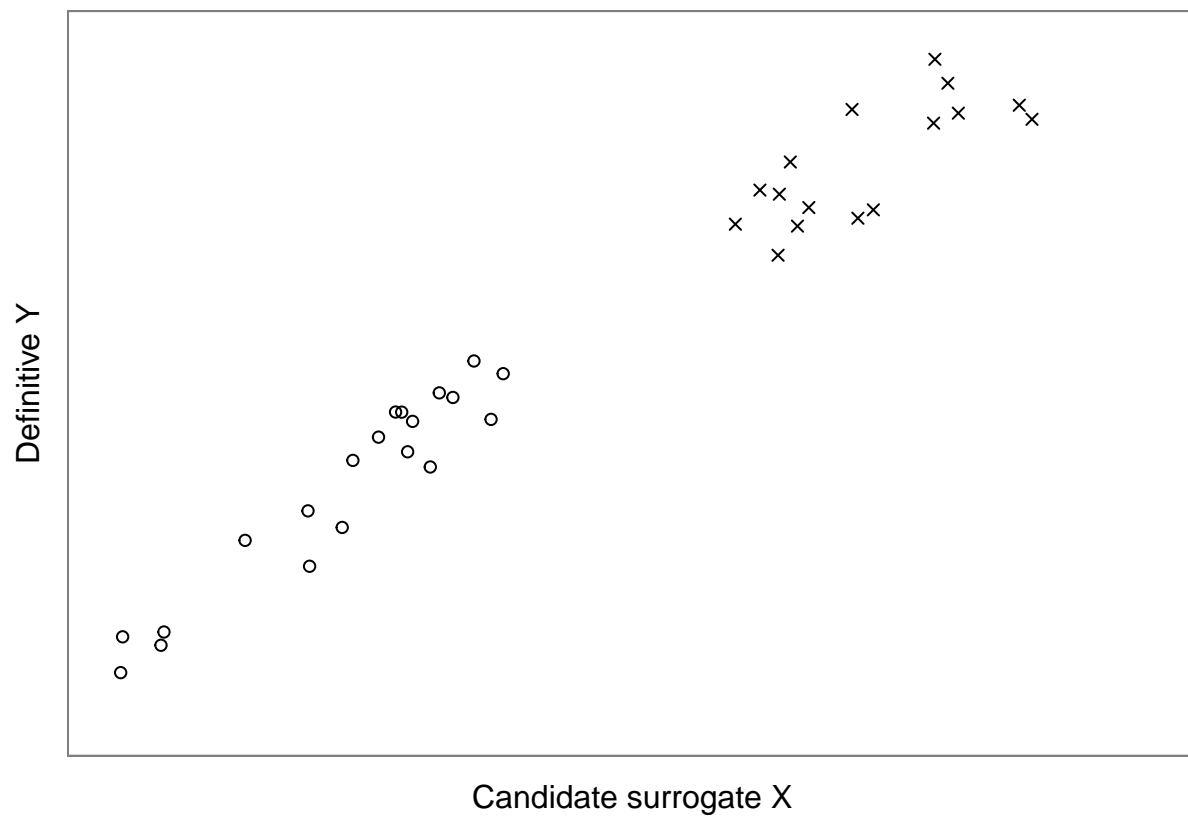
2015-07-29

Figure 1

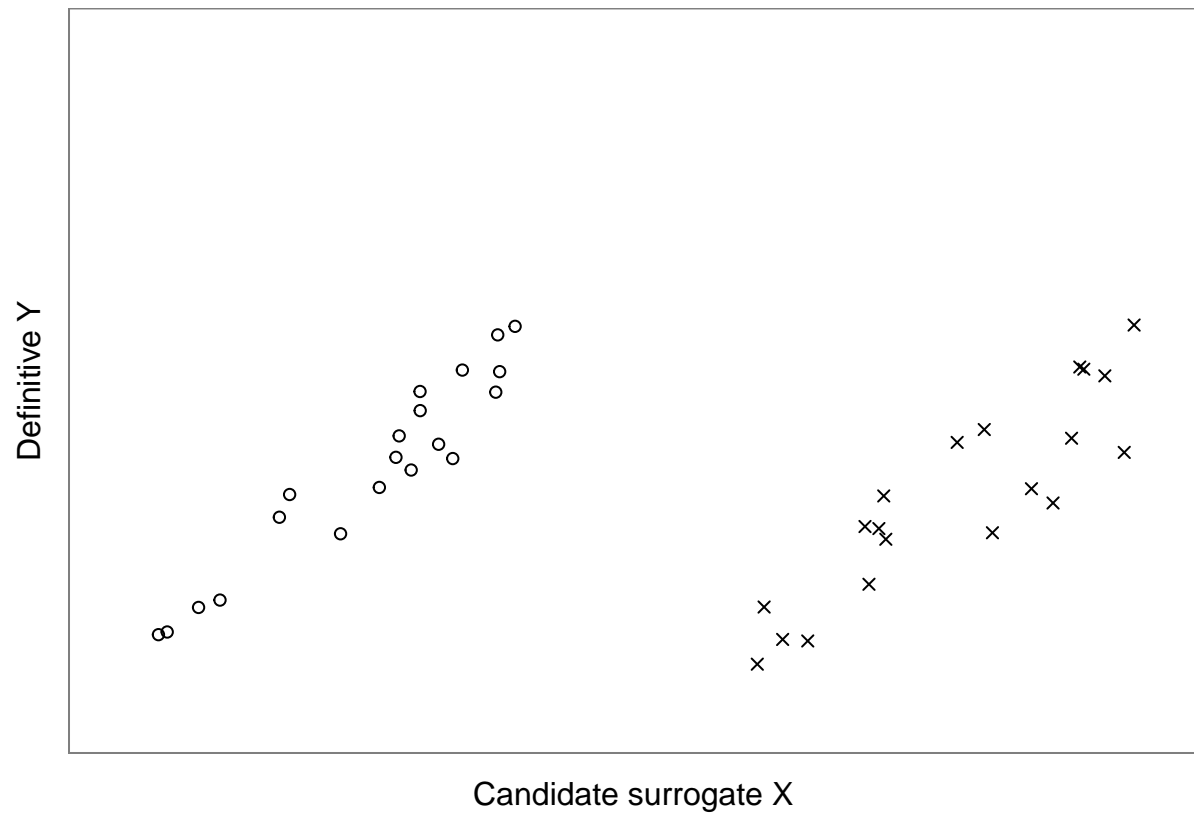
```
library(pcrmeta)
```

```
plot_surrogates(TRUE, TRUE)
```

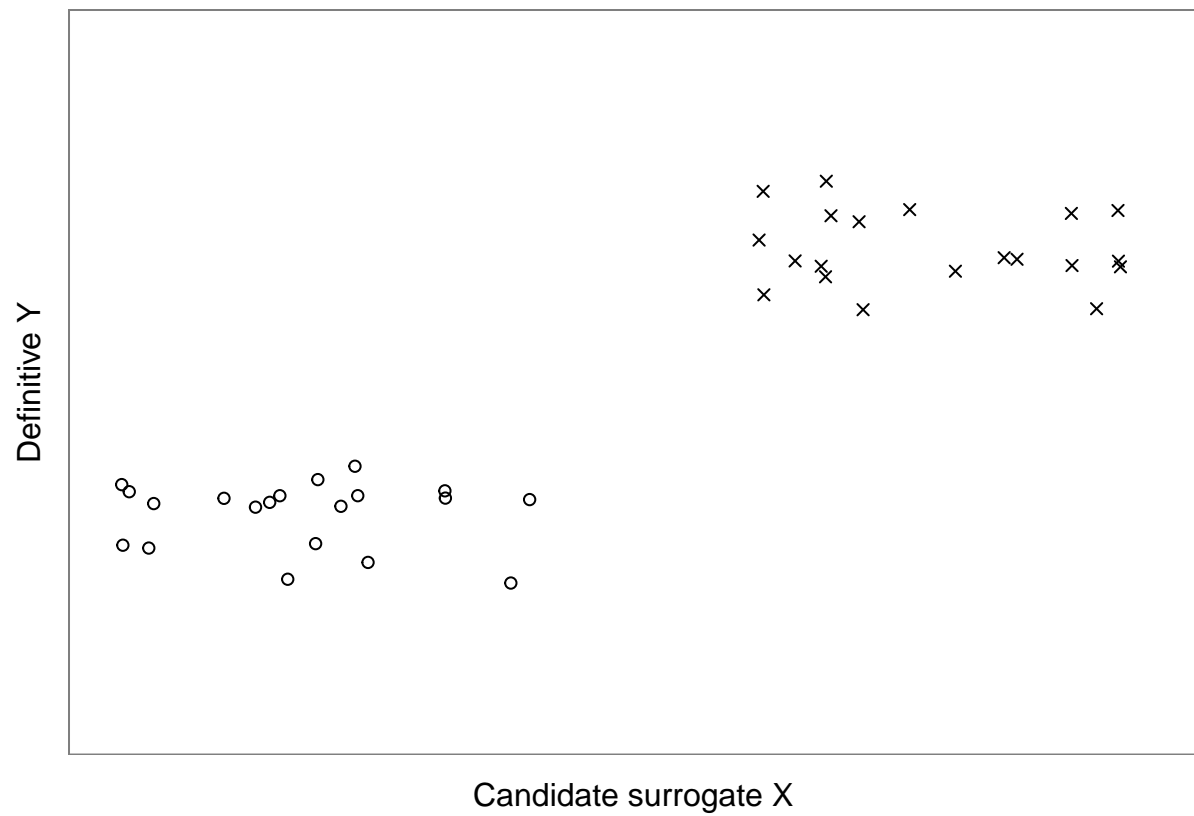
```
## Warning: Removed 4 rows containing missing values (geom_point).
```



```
plot_surrogates(TRUE, FALSE)
```



```
plot_surrogates(FALSE, TRUE)
```



Data

This is an attempt to reanalyze the data from the meta analyses described in Cortazar et al. (2014). Cortazar et al. (2014) includes the results from 10 trials in their trial-level analysis. Some of the comparisons are not included in the trial-level analysis because they were randomized and some were single-arm trials.

	LancetID	Patients	OR.PCR	HR.DFS	HR.OS	PCR.SE	DFS.SE	OS.SE	Trial
14	A	950	1.0316	1.0472	0.8105	0.1559	0.1322	0.1715	GeparQuatro1
15	B	913	2.1002	1.1153	1.0186	0.2306	0.1180	0.1618	GeparDuo
16	C	942	0.9608	0.8795	0.8396	0.1605	0.1357	0.1771	GeparQuatro2
17	D	1856	1.1110	0.8396	0.8900	0.1704	0.0772	0.1069	EORTC
18	E	733	1.4997	0.8688	0.7994	0.2040	0.1431	0.1948	PREPARE
19	F	2411	2.2416	0.8877	0.9192	0.1364	0.0819	0.1035	NSABP
20	G	1390	1.2510	0.7494	0.7609	0.1292	0.0987	0.1451	GeparTrio1
21	H	622	0.8806	0.6906	0.8203	0.3481	0.1287	0.2049	GeparTrio2
22	I	671	2.1719	0.7092	0.6904	0.2323	0.1378	0.1825	AGO
23	J	235	3.0383	0.6689	0.5999	0.3002	0.2177	0.3066	NOAH

We have included an additional 13 trials that were analyzed by Berruti et al. (2014). That paper included 29 trials total, but we excluded the small ones (less than 20 events), and ones that did not have sufficient information for the analysis.

	LancetID	Patients	OR.PCR	HR.DFS	HR.OS	PCR.SE	DFS.SE	OS.SE	Trial
	NA	199	1.528	0.713	0.892	0.459	0.218	0.217	Arun
	NA	150	1.585	0.773	0.871	0.928	0.248	0.295	Bald
	NA	96	3.000	0.413	0.397	0.701	0.399	0.925	Chen
	NA	451	1.003	1.180	1.410	0.310	0.195	0.252	Chua
	NA	211	3.165	0.730	0.770	0.830	0.233	0.273	Cocc
	NA	372	1.232	1.030	1.190	0.255	0.158	0.196	Ellis
	NA	200	2.940	0.723	0.644	0.504	0.203	0.243	Fraci
	NA	209	2.472	0.967	0.180	0.411	0.392	0.791	Lee
	NA	363	0.716	0.818	0.818	0.317	0.157	0.171	Mansi
	NA	426	1.021	1.050	0.760	0.368	0.161	0.203	Smith
	NA	448	0.677	1.051	1.010	0.368	0.120	0.136	Theras
	NA	477	0.928	0.907	0.671	0.215	0.256	0.408	Toi
	NA	89	0.661	0.900	1.260	0.529	0.437	0.540	Walker

Methods

Here we are going to fit model (7) from Korn, Albert, and McShane (2005). This is using the reported standard errors on the log OR and log HR, instead of the variances of the arm-specific summaries. The model is

$$\log(OR_i) = \mu + m_i + \epsilon_i$$

$$\log(HR_i) = \alpha + \beta * (\mu + m_i) + g_i + \delta_i$$

where for trial i , OR_i is the observed odds ratio comparing pCR for the two treatment arms, HR_i is the observed hazard ratio comparing the treatment arms, $\mu + m_i$ is the true log odds ratio for pCR, μ is a fixed

effect representing the average log odds ratio across trials, and m_i is a random effect with mean 0 and variance σ_m^2 , ϵ_i is a random error with standard deviation equal to standard error of the estimate of the log odds ratio. In the second equation HR_i is the observed hazard ratio comparing two treatments for trial i . This equation specifies a linear relationship, with intercept α and slope β between the true log hazard ratio and the true log odds ratio for pCR. Here g_i is a random effect with mean 0 and variance σ_g^2 and δ_i is a random error with mean 0 and standard deviation equal to the standard error of the estimate of the log hazard ratio.

This model implies a bivariate normal likelihood for OR_i and HR_i . The likelihood of this model is coded as follows:

```
like.hood.model4
```

```
## function(Yeff, Xeff, Yse, Xse, param){
##
##   stopifnot(length(param) == 5)
##   stopifnot(length(Xeff) == length(Yeff))
##   stopifnot(length(Yse) == length(Yeff))
##   stopifnot(length(Xse) == length(Xeff))
##
##   vmg <- param[5]
##   beta <- param[3]
##   muy <- param[2]
##   vmx <- param[4]
##   mux <- param[1]
##   n <- length(Yeff)
##
##   c11 <- vmx + Xse^2
##   c22 <- beta^2 * vmx + vmg + Yse^2
##   c12 <- beta * vmx
##
##   sigma <- rbind(cbind(diag(c11), diag(c12, nrow = n)),
##                  cbind(diag(c12, nrow = n), diag(c22)))
##
##   XX <- c(Xeff, Yeff)
##   mu = c(rep(mux, n), rep(muy + beta * mux, n))
##
##   hood <- mvtnorm::dmvnorm(XX, mean = mu, sigma = sigma, log = TRUE)
##
##   -hood
##
## }
## <environment: namespace:pcrmeta>
```

The estimates are obtained by maximum likelihood, with model based standard errors. Due to the small sample size, the maximum likelihood estimates of the variance components will be on average too small. General procedures for handling this problem (e.g., REML), are not readily applicable in the measurement error model being considered here. Instead, we have used an ad hoc adjustment of multiplying $\hat{\sigma}_m^2$ by $n/(n-1)$ and $\hat{\sigma}_g^2$ by $n/(n-3)$. The adjusted MLEs are denoted $\tilde{\sigma}_g^2$ and $\tilde{\sigma}_m^2$.

For a new trial with $i = 0$ in which pCR is assessed but not the definitive outcome, it is of interest to predict the true hazard ratio for a new trial based on its observed pCR results and sample size. Let $\log(OR_0) = v$ be the observed log odds ratio and σ_0^2 be the variance of v . The quantity of interest is

$$\Delta = E[\alpha + \beta(\mu + m_0) + g_0 | \mu + m_0 + \epsilon_0 = v] =$$

$$\alpha + \beta\mu + E[m_0 | \epsilon_0 = v - \mu] =$$

$$\alpha + \beta v \left(\frac{\sigma_m^2}{\sigma_m^2 + \sigma_0^2} \right) + \beta\mu \left(1 - \frac{\sigma_m^2}{\sigma_m^2 + \sigma_0^2} \right)$$

\$\$

The estimator of Δ , $\tilde{\Delta}$ is obtained by plugging in the adjusted maximum likelihood estimators. The variance of $\tilde{\Delta}$ is obtained using the delta method.

To obtain a prediction interval that contains the true log hazard ratio for a new trial, that is, that contains $\alpha + \beta(\mu + m_0) + g_0$, let

$$V = Var(\tilde{\Delta}) + Var(g_0) + \beta^2 \sigma_m^2 \left(1 - \frac{\sigma_m}{\sqrt{\sigma_m^2 + \sigma_0^2}} \right)^2,$$

and let \tilde{V} denote its plug-in estimate. Then an approximate 95% prediction interval for the true log hazard ratio is

$$\tilde{\Delta} \pm t_{n-3} \sqrt{\tilde{V}}.$$

The code to estimate these predictions and standard errors (using the delta method) is as follows:

```
predict_klmfit
```

```
## function(fit4, v, sig0){
##
##   delta <- fit4$par["muy"] + fit4$par["beta"] * v * (fit4$par["vmx"] / (fit4$par["vmx"] + sig0)) +
##     fit4$par["beta"] * fit4$par["mux"] * (1 - fit4$par["vmx"] / (fit4$par["vmx"] + sig0))
##
##   ## delta method
##
##   mu.deriv <- deriv(expression(muy + beta * v * (vmx/(vmx + sig0)) +
##                                beta * mux * (1 - vmx/(vmx + sig0))),
##                      namevec = c("mux", "muy", "beta", "vmx", "vmg"),
##                      function.arg = function(mux, muy, beta, vmx, vmg, v, sig0){} )
##
##   delta <- do.call(mu.deriv, as.list(c(fit4$par, v, sig0)))
##   mu.grad <- attr(delta, "gradient")
##   var.delta <- c(mu.grad %*% solve(fit4$hessian) %*% t(mu.grad))
##
##   a3 <- fit4$par["beta"]^2 * fit4$par["vmx"] *
##     (1 - sqrt(fit4$par["vmx"])/sqrt(fit4$par["vmx"] + sig0))
##
##   tilde.v <- var.delta + fit4$par["vmg"] + a3
##
##
##   list(delta = delta, se.delta = sqrt(var.delta), se.pred = sqrt(tilde.v), a3 = a3)
## }
## <environment: namespace:pcrmeta>
```

The performance of this method, including prediction interval coverage, is assessed in the simulation study vignette.

Results

Event Free Survival

Ten trials only

```
ten.dfs <- with(trials10, fit_model4(log(HR.DFS), log(OR.PCR), DFS.SE, PCR.SE))
kable(ten.dfs$summary)
```

	Est	SE
mux	0.3873	0.1198
muy	-0.1828	0.0805
beta	0.0452	0.1836
vmx	0.1119	0.0730
vmg	0.0075	0.0152

Prediction table

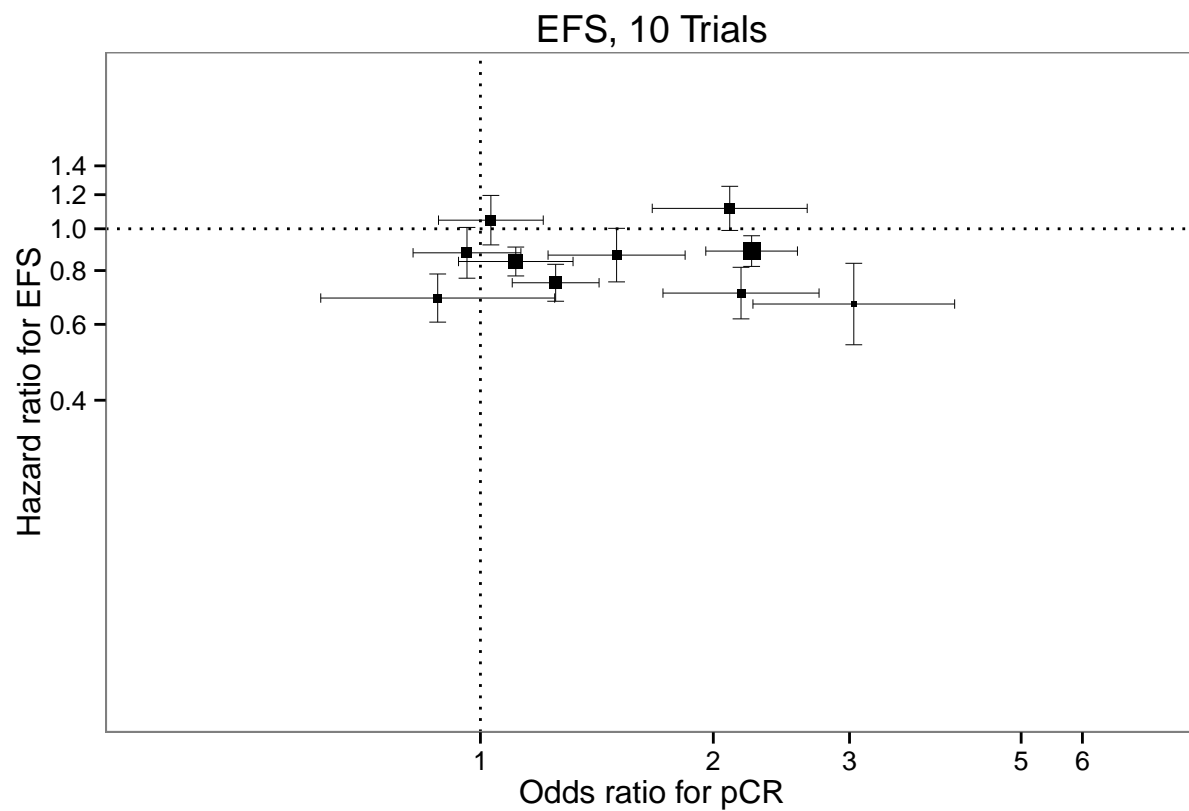
```
library(magrittr)
library(tidyr)

params <- data.frame(p1 = rep(1:4/10, 3),
                     p0 = rep(c(.1, .1, .1, .2), 3),
                     n = sort(rep(c(100, 300, 1000), 4)))

params$CI <- sapply(1:nrow(params), function(i){
  with(params[i, ], fill_table(p1, p0, n, ten.dfs)$a1)
})

params %>%
  spread(n, CI) %>% kable
```

p1	p0	100	300	1000
0.1	0.1	0.84 (0.67, 1.07)	0.84 (0.66, 1.07)	0.84 (0.65, 1.08)
0.2	0.1	0.85 (0.67, 1.08)	0.86 (0.67, 1.10)	0.86 (0.65, 1.13)
0.3	0.1	0.86 (0.66, 1.11)	0.87 (0.63, 1.19)	0.88 (0.58, 1.32)
0.4	0.2	0.86 (0.67, 1.10)	0.86 (0.65, 1.15)	0.87 (0.63, 1.20)



All 23 trials

```
all.dfs <- with(rbind(trials10, trials13),
  fit_model4(log(HR.DFS), log(OR.PCR), DFS.SE, PCR.SE))
kable(all.dfs$summary)
```

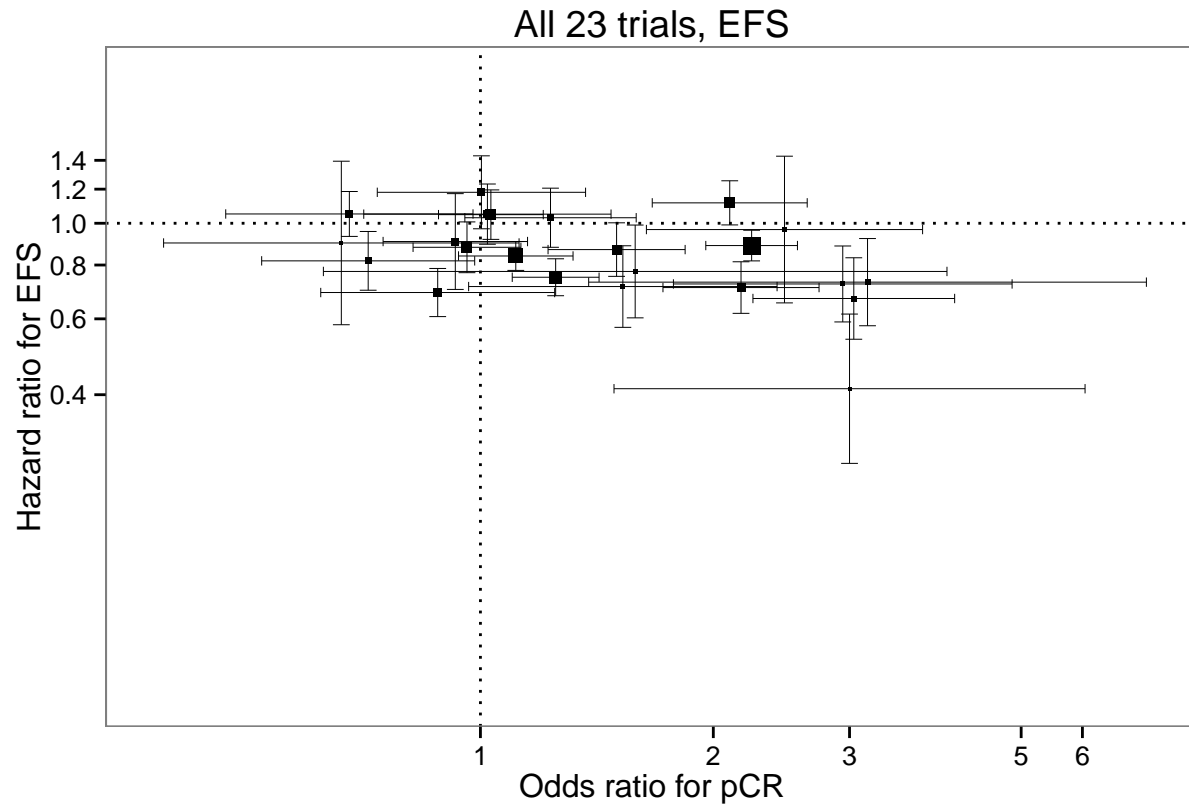
	Est	SE
mux	0.3057	0.0942
muy	-0.1072	0.0582
beta	-0.1230	0.1506
vmx	0.1120	0.0610
vmg	0.0076	0.0093

Prediction table

```
params$CI <- sapply(1:nrow(params), function(i){
  with(params[i, ], fill_table(p1, p0, n, all.dfs)$a1)
})

params %>%
  spread(n, CI) %>% kable
```

p1	p0	100	300	1000
0.1	0.1	0.87 (0.71, 1.07)	0.88 (0.71, 1.08)	0.89 (0.72, 1.10)
0.2	0.1	0.85 (0.69, 1.05)	0.84 (0.67, 1.05)	0.83 (0.65, 1.04)
0.3	0.1	0.84 (0.67, 1.05)	0.81 (0.62, 1.06)	0.78 (0.56, 1.09)
0.4	0.2	0.84 (0.67, 1.05)	0.82 (0.64, 1.05)	0.81 (0.62, 1.06)



Overall Survival

Ten trials only

```
ten.os <- with(trials10, fit_model4(log(HR.OS), log(OR.PCR), OS.SE, PCR.SE))
```

```
## Warning in sqrt(diag(solve(fit4$hessian))): NaNs produced
```

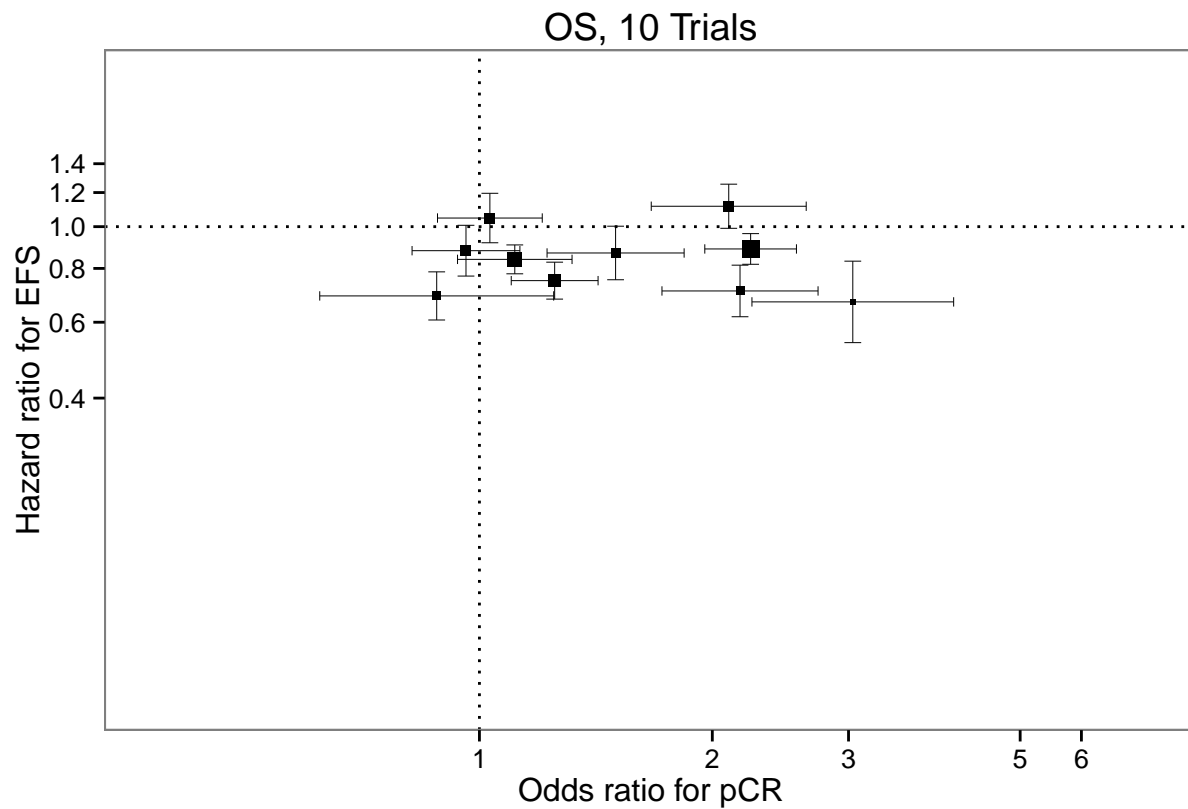
```
kable(ten.os$summary)
```

	Est	SE
mux	0.3882	0.1188
muy	-0.1866	0.0827
beta	0.0525	0.1607
vmx	0.1100	0.0718
vmg	0.0000	NaN

Prediction table

```
params$CI <- sapply(1:nrow(params), function(i){
  with(params[i, ], fill_table(p1, p0, n, ten.os)$a1)
})
params %>%
  spread(n, CI) %>% kable
```

p1	p0	100	300	1000
0.1	0.1	0.84 (0.75, 0.94)	0.84 (0.74, 0.96)	0.83 (0.71, 0.98)
0.2	0.1	0.85 (0.77, 0.94)	0.86 (0.77, 0.95)	0.86 (0.75, 0.98)
0.3	0.1	0.86 (0.76, 0.97)	0.87 (0.72, 1.05)	0.88 (0.67, 1.16)
0.4	0.2	0.86 (0.77, 0.95)	0.86 (0.75, 1.00)	0.87 (0.72, 1.05)



All 23 trials

```
all.os <- with(rbind(trials10, trials13),
  fit_model4(log(HR.OS), log(OR.PCR), OS.SE, PCR.SE))
kable(all.os$summary)
```

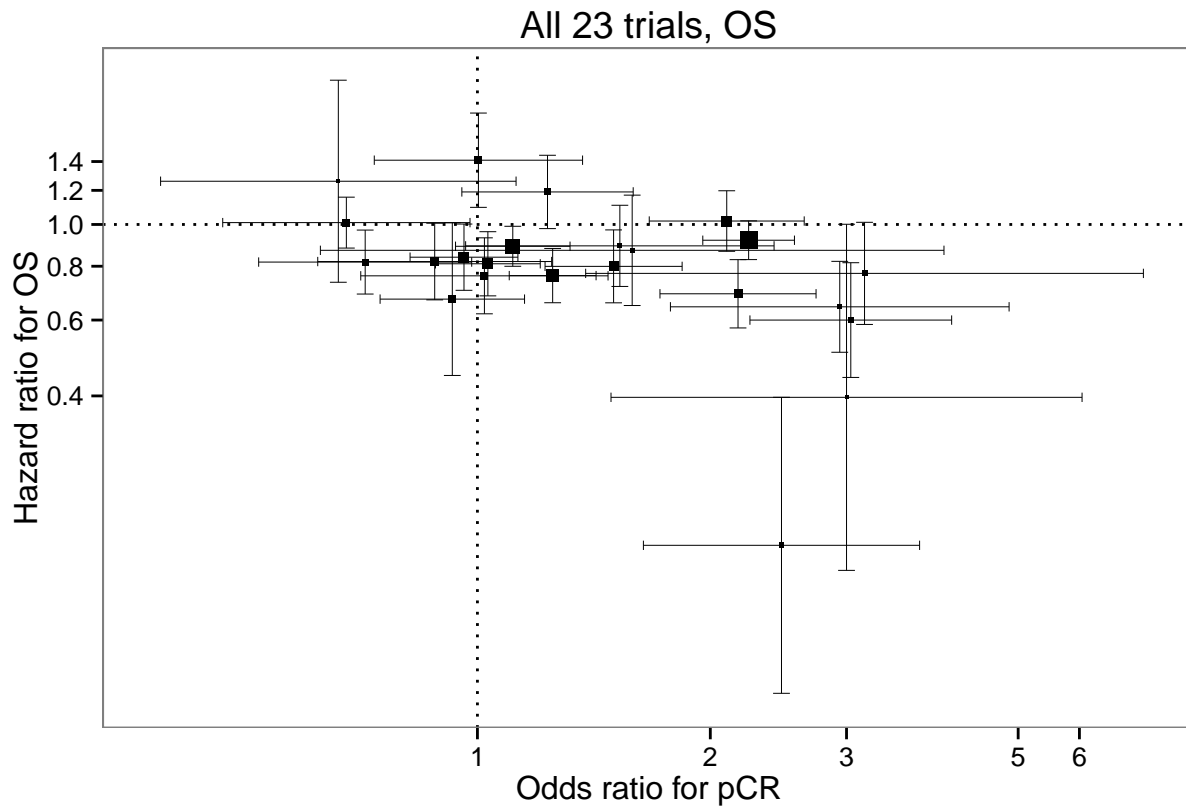
	Est	SE
mux	0.3011	0.0937
muy	-0.1214	0.0575
beta	-0.0740	0.2146
vmx	0.1110	0.0611
vmg	0.0000	0.0381

Prediction table

```
params$CI <- sapply(1:nrow(params), function(i){
  with(params[i, ], fill_table(p1, p0, n, all.os)$a1)
})

params %>%
  spread(n, CI) %>% kable
```

p1	p0	100	300	1000
0.1	0.1	0.87 (0.77, 0.98)	0.87 (0.79, 0.97)	0.88 (0.79, 0.98)
0.2	0.1	0.86 (0.73, 1.01)	0.85 (0.69, 1.05)	0.84 (0.64, 1.10)
0.3	0.1	0.85 (0.68, 1.06)	0.83 (0.60, 1.16)	0.82 (0.52, 1.28)
0.4	0.2	0.85 (0.69, 1.05)	0.84 (0.63, 1.11)	0.83 (0.59, 1.17)



References

- Berruti, Alfredo, Vito Amoroso, Fabio Gallo, Valentina Bertaglia, Edda Simoncini, Rebecca Pedersini, Laura Ferrari, Alberto Bottini, Paolo Bruzzi, and Maria Pia Sormani. 2014. “Pathologic Complete Response as a Potential Surrogate for the Clinical Outcome in Patients with Breast Cancer After Neoadjuvant Therapy: A Meta-Regression of 29 Randomized Prospective Studies.” *Journal of Clinical Oncology* 32 (34). American Society of Clinical Oncology: 3883–91.
- Cortazar, Patricia, Lijun Zhang, Michael Untch, Keyur Mehta, Joseph P Costantino, Norman Wolmark, Hervé Bonnefoi, et al. 2014. “Pathological Complete Response and Long-Term Clinical Benefit in Breast Cancer: The CTNeoBC Pooled Analysis.” *The Lancet* 384 (9938). Elsevier: 164–72.
- Korn, Edward L, Paul S Albert, and Lisa M McShane. 2005. “Assessing Surrogates as Trial Endpoints Using Mixed Models.” *Statistics in Medicine* 24 (2). Wiley Online Library: 163–82.