

# Phaco meta analysis

*Patrick Mineault*

*14/02/2017*

## Load data

We load data from a CSV exported from Stata. The Mo variables refer to what happens after 6 months. The W, X, Y, Z, AA variables refer to what happens after 12 months. That's a bug in how Stata exports names of variables which start with a number - the columns were named 6mo... and 12mo....

```
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages -----
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
library(meta)
```

```
## Loading 'meta' package (version 4.7-0).
## Type 'help("meta-package")' for a brief overview.
```

```
library(ggrepel)
```

```
setwd("~/Documents/phaco")
df <- read.csv("phaco.csv", na.strings='-')
df <- df %>% rename(SixMoEyes = MoEyes,
                   SixMoIOPMean = MoIOPMean,
                   SixMoIOPStdDev = MoIOPStdDev,
                   SixMoAbsIOPChangeMean = MoAbsIOPChangeMean,
                   SixMoAbsIOPChangeStdDev = MoAbsIOPChangeStdDev,
                   OneYEyes = W,
                   OneYIOPMean = X,
                   OneYIOPStdDev = Y,
                   OneYAbsIOPChangeMean = Z,
                   OneYAbsIOPChangeStdDev = AA,
                   LastPeriodAbsIOPChangeStdDev = LastPeriodAbsIOPChangeStd,
                   LastPeriodEyes = LastPeriodofEyes
)

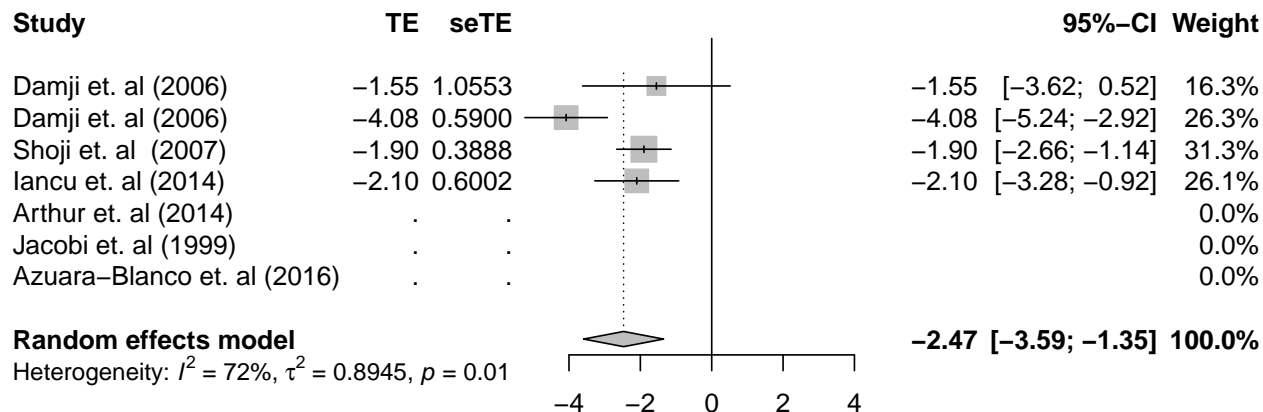
df <- df %>% mutate(subtype = as.factor(ifelse(is.na(OAG), 'OAG',
                                              ifelse(OAG > 50, 'OAG',
                                                    ifelse(ACG > 50, 'ACG',
                                                          ifelse(PXG > 50, 'PXG', NA))))))

df <- df %>% mutate(study.name = paste0(Author, ' (' , Year, ')'))
```

# Analysis without imputation

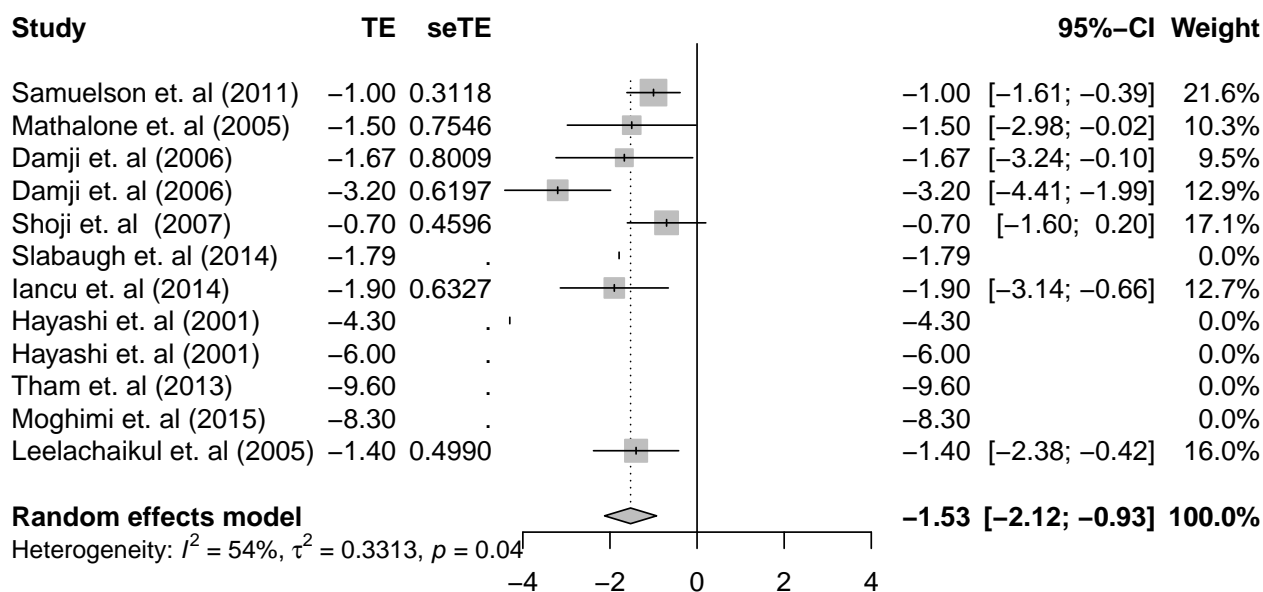
## 6 month follow-up

```
m <- metagen(SixMoAbsIOPChangeMean,
              SixMoAbsIOPChangeStdDev / sqrt(SixMoEyes),
              study.name,
              data=df,
              subset=!is.na(df$SixMoEyes))
forest(m, comb.fixed=FALSE)
```



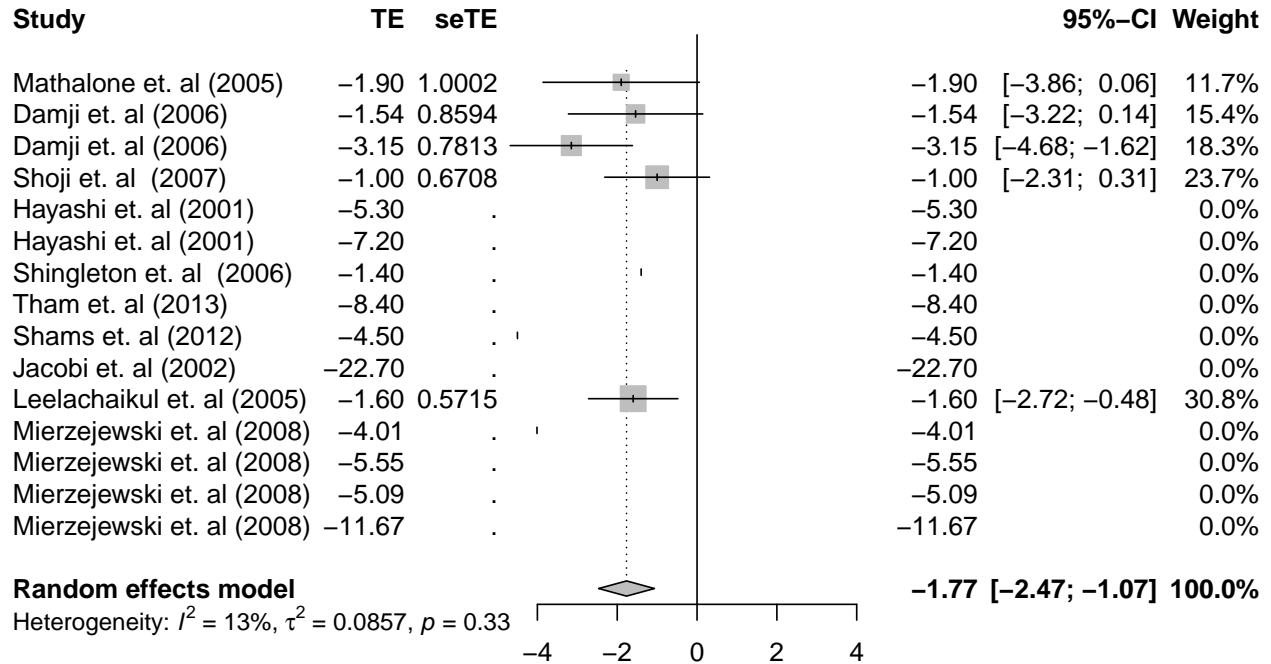
## 12-month follow up

```
m <- metagen(OneYAbsIOPChangeMean,
              OneYAbsIOPChangeStdDev / sqrt(OneYEyes),
              study.name,
              data=df,
              subset=!is.na(df$OneYAbsIOPChangeMean))
forest(m, comb.fixed = FALSE)
```



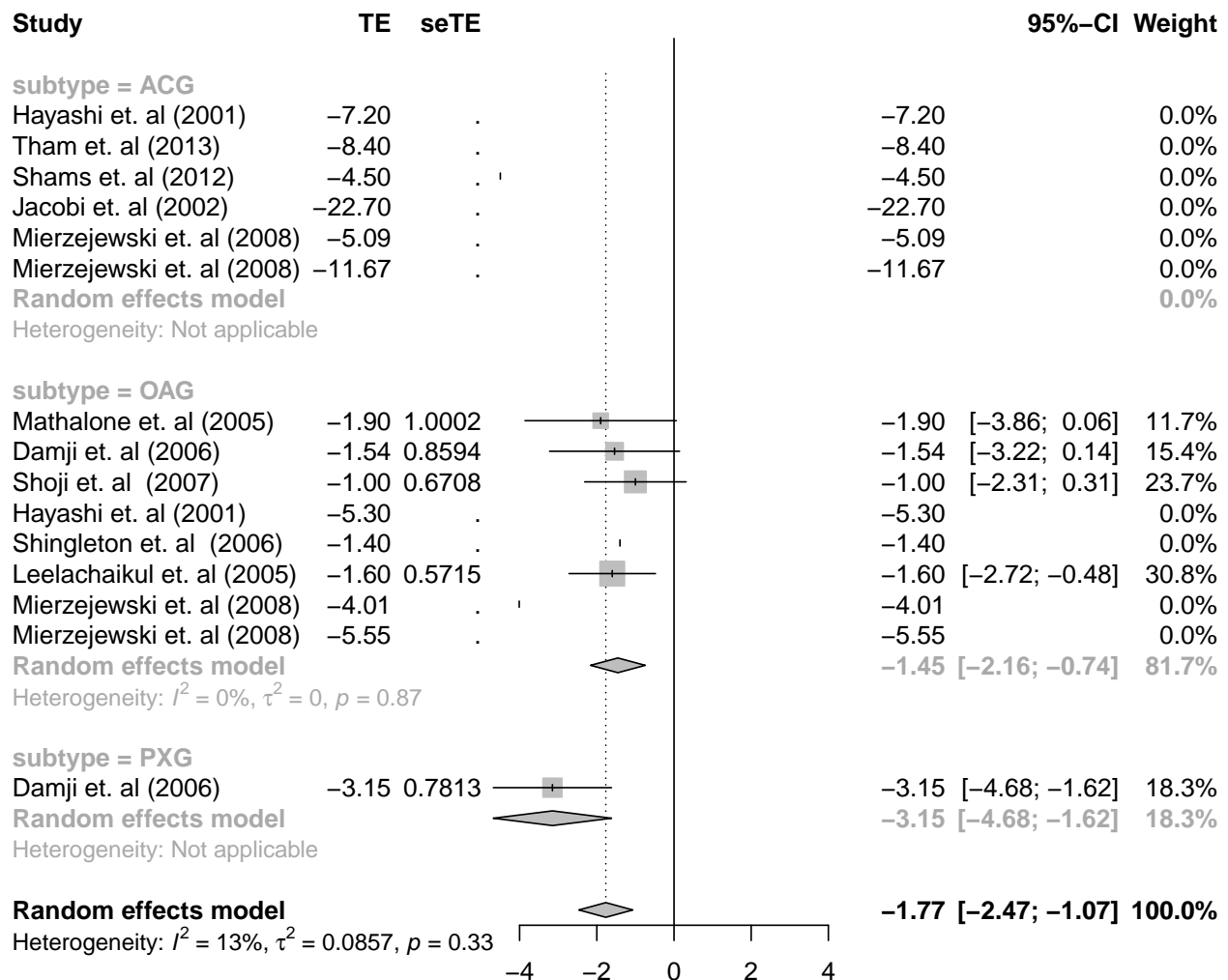
```
## Last period
```

```
m <- metagen(LastPeriodAbsIOPChangeMean,
             LastPeriodAbsIOPChangeStdDev / sqrt(LastPeriodEyes),
             study.name,
             data=df,
             subset=!is.na(df$LastPeriodAbsIOPChangeMean))
forest(m, comb.fixed = FALSE)
```



## Subgroup analysis

```
m <- metagen(LastPeriodAbsIOPChangeMean,
             LastPeriodAbsIOPChangeStdDev / sqrt(LastPeriodEyes),
             study.name,
             data=df,
             subset=!is.na(df$LastPeriodAbsIOPChangeMean),
             byvar=subtype)
forest(m, comb.fixed = FALSE)
```



## Analyses with imputation

I'm going to start by imputing on the basis of a low loss in follow up, high correlation between pre and post measures, and no change in relative IOP for those patients that were lost in follow up.

```
# Consider two range of loss of follow up.
losses <- list(lo=c(.94, .91, .87),
               hi=c(.82, .72, .51))

# Assume a range of correlations between PreOp and PostOp periods.
corrs <- list(lo=c(.25, .25, .25),
              hi=c(.45, .45, .45))

# And a range of outcomes for the missing eyes.
deltas <- list(lo=c(0, 0, 0), # Same outcome in the unseen arm.
               hi=c(5, 5, 5)) # Worse outcome in the unseen arm.

infer.mean.sem <- function(N.b, m.b, s.b, N.a, m.a, s.a, m.d, s.d, rho, m.delta) {
  # Infer the mean and SEM of the deviation in the metric measured before and after the intervention.
  # N.b, m.b, s.b: N, mean, S.D. of metric before the intervention
```

```

# N.a, m.a, s.a: N, mean, S.D. of metric after the intervention
# m.d, s.d: mean, S.D. of metric after intervention minus the metric before the intervention for
# the group of survivors (N.a). Can be NA.
# rho: Assumed correlation between before and after scores. Will be used to infer s.d. if s.d is NA
# m.delta: the assumed mean delta between the m.b for the non-survivors minus the survivors.
# Use equations in Section 6.1 of Schwarzer, Carpenter & Rucker (2014), Meta-Analysis with R
m.d <- ifelse(is.na(m.d), m.a - m.b, m.d)
s.d <- ifelse(is.na(s.d), sqrt(s.a ** 2 + s.b ** 2 - 2 * rho * s.a * s.b), s.d)

# Now generalize the mean difference from the observed subset to the full dataset.
# pi := unobserved portion
pi = (N.b - N.a) / N.b
m.d.full <- m.d + pi * m.delta
# Let's assume that delta ~ N(m.delta, s.d)
sem.d.full <- sqrt(1 + pi ** 2) * s.d / sqrt(N.a)
return(data.frame(m=m.d.full, sem=sem.d.full))
}

library(testthat)

##
## Attaching package: 'testthat'

## The following object is masked from 'package:dplyr':
##
## matches

## The following object is masked from 'package:purrr':
##
## is_null

expect_equal(infer.mean.sem(25, 0, 1, 25, 0, 1, NA, NA, 0, 0)$sem, sqrt(2)/sqrt(25))
expect_equal(infer.mean.sem(25, 0, 1, 25, 0, 1, NA, NA, 0.5, 0)$sem, 1/sqrt(25))
expect_equal(infer.mean.sem(25, 0, 1, 25, 0, 1, NA, NA, 0, 0)$m, 0)
expect_equal(infer.mean.sem(25, 0, 1, 16, 0, 1, NA, NA, 0, 0)$sem, sqrt(1 + (9 / 25) ** 2) * sqrt(2)/sqrt(25))
expect_equal(infer.mean.sem(25, -1, 1, 16, 2, 1, NA, NA, 0, 0)$m, 3)
expect_equal(infer.mean.sem(25, -1, 1, 16, 2, 1, NA, NA, 0, 0)$m, 3)
expect_equal(infer.mean.sem(20, 5, 1, 20, -5, 1, NA, NA, 0, 10)$m, -10)
expect_equal(infer.mean.sem(20, 5, 1, 10, -5, 1, NA, NA, 0, 20)$m, 0)

impute.df <- function(df, loss, corr, delta) {
  # Impute missing data using the bone-headed method of just assuming the mean effect of
  # - loss of follow-up
  # - correlation between Pre and Post IOP metrics
  # - delta between IOP of eyes that were lost in follow up and eyes that were ok.
  df <- df %>% mutate(imp.SixMoEyes = ifelse(is.na(SixMoEyes), round(loss[1] * PreOpEyes), SixMoEyes),
    imp.OneYEyes = ifelse(is.na(OneYEyes), round(loss[2] * PreOpEyes), OneYEyes),
    imp.LastPeriodEyes = ifelse(is.na>LastPeriodEyes), round(loss[3] * PreOpEyes), LastPeriodEyes))

  df_ <- with(df, infer.mean.sem(PreOpEyes, PreOpIOPMean, PreOpIOPStdDev,
    imp.SixMoEyes, SixMoIOPMean, SixMoIOPStdDev,
    SixMoAbsIOPChangeMean, SixMoAbsIOPChangeStdDev, corr[1], delta[1]))

  df$imp.SixMoIOPChangeMean <- df_$m
  df$imp.SixMoIOPChangeSEM <- df_$sem

```

```

df_ <- with(df, infer.mean.sem(PreOpEyes, PreOpIOPMean, PreOpIOPStdDev,
                              imp.OneYEyes, OneYIOPMean, OneYIOPStdDev,
                              OneYAbsIOPChangeMean, OneYAbsIOPChangeStdDev, corr[2], delta[2]))
df$imp.OneYIOPChangeMean <- df_$m
df$imp.OneYIOPChangeSEM <- df_$sem

df_ <- with(df, infer.mean.sem(PreOpEyes, PreOpIOPMean, PreOpIOPStdDev,
                              imp.LastPeriodEyes, LastPeriodIOPMean, LastPeriodIOPStdDev,
                              LastPeriodAbsIOPChangeMean, LastPeriodAbsIOPChangeStdDev, corr[3], del

df$imp.LastPeriodIOPChangeMean <- df_$m
df$imp.LastPeriodIOPChangeSEM <- df_$sem

# Patch up NAs for std dev of medications.
df$imp.RxPreOpStdDev <- ifelse(is.na(df$RxPreOpStdDev),
                              .5* df$RxPreOpMean + .2,
                              pmax(df$RxPreOpStdDev, .2))

df$imp.RxPostOpStdDev <- ifelse(is.na(df$RxPostOpStdDev),
                              .5* df$RxPostOpMean + .2,
                              pmax(df$RxPostOpStdDev, .2))

return(df)
}

# Q: are we dealing properly with loss of follow-up means?
# Verify where available if the before, after measurements, and changes match up.

df <- impute.df(df, losses[['lo']], corrs[['hi']], deltas[['lo']])

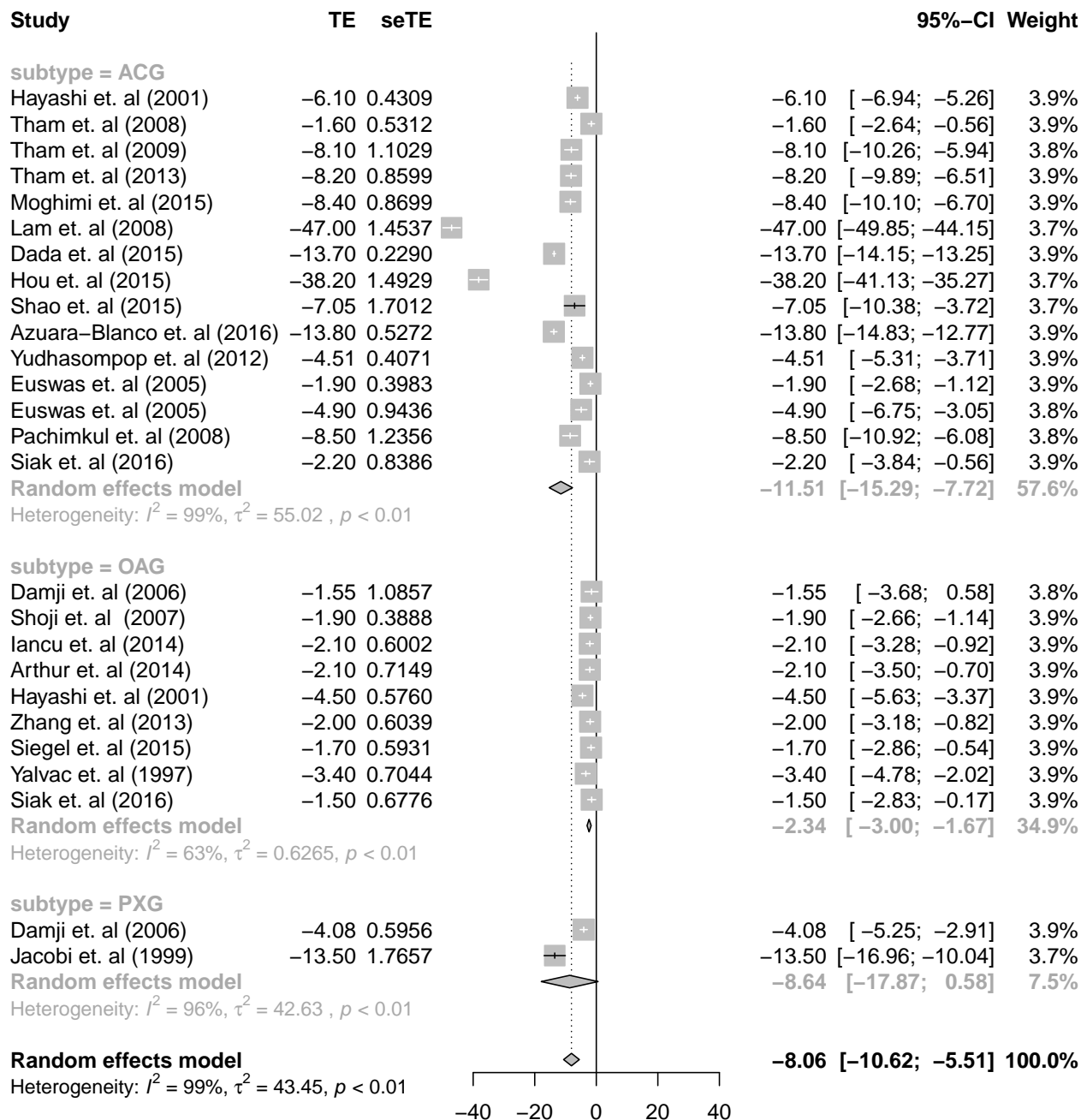
```

## 6 month follow-up

```

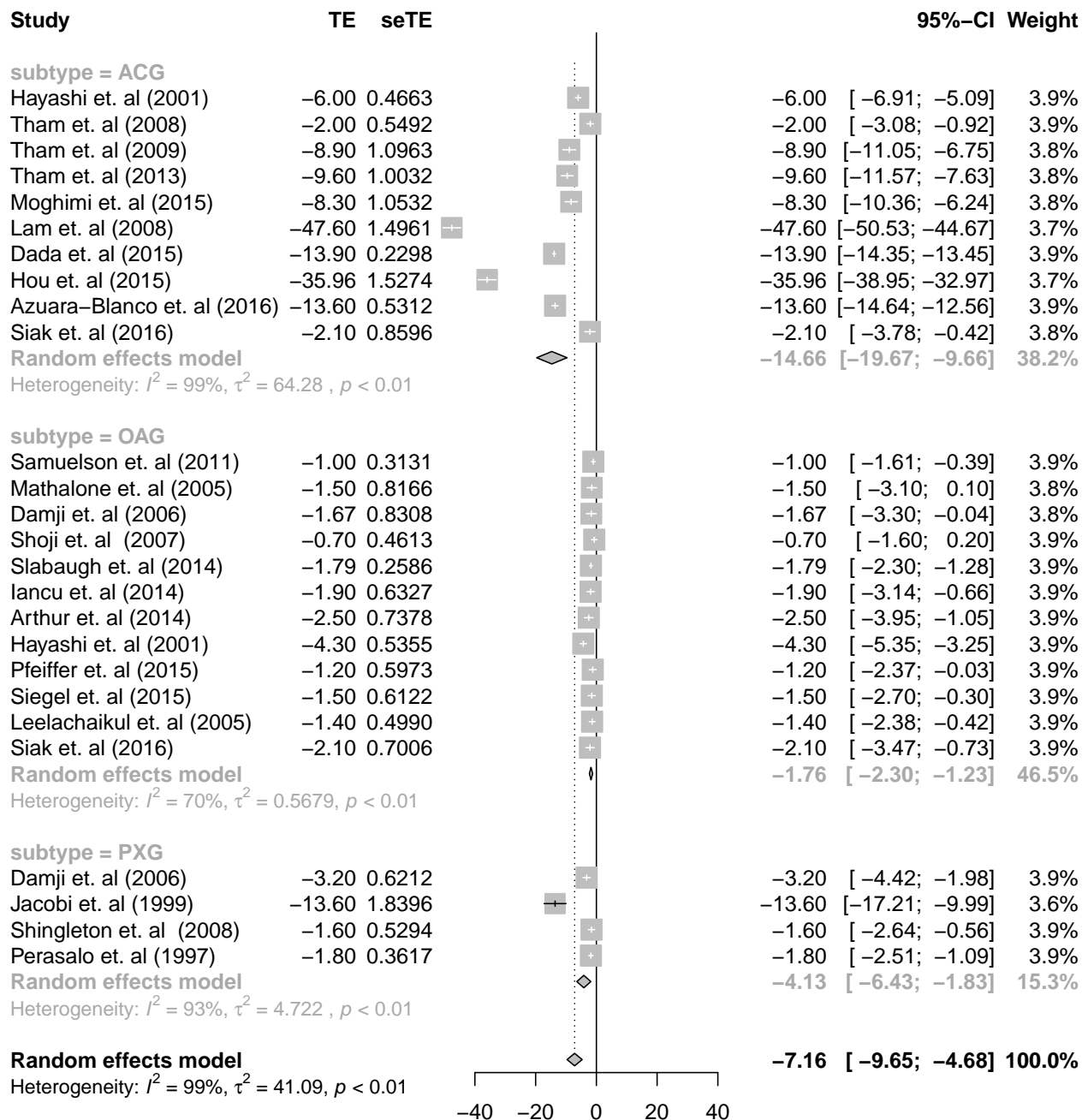
m <- metagen(imp.SixMoIOPChangeMean,
             imp.SixMoIOPChangeSEM,
             study.name,
             data=df,
             subset=!is.na(imp.SixMoIOPChangeSEM),
             byvar=subtype)
forest(m, comb.fixed=FALSE)

```



## 12-month follow up

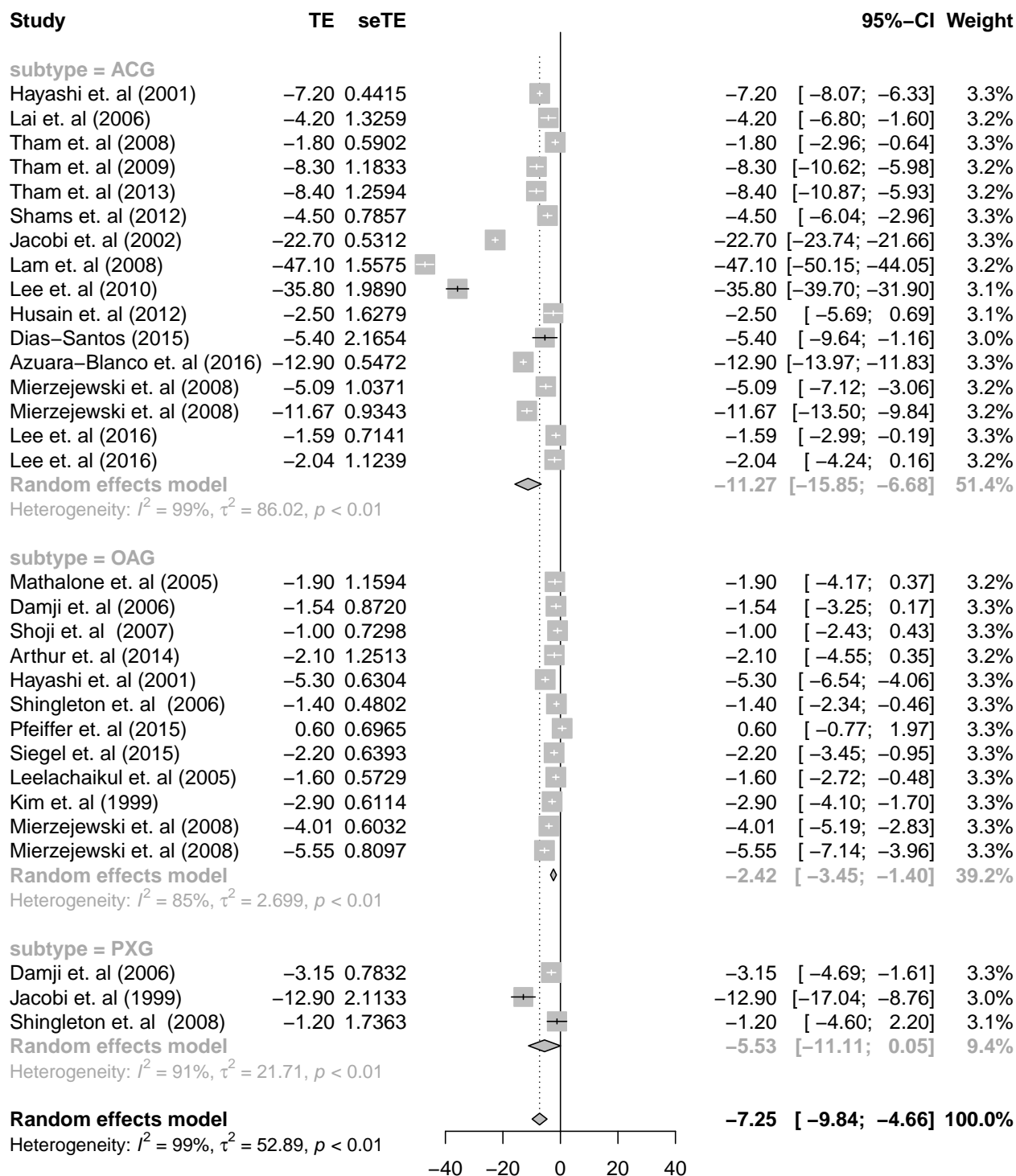
```
m <- metagen(imp.OneYIOPChangeMean,
              imp.OneYIOPChangeSEM,
              study.name,
              data=df,
              subset=!is.na(imp.OneYIOPChangeSEM),
              byvar=subtype)
forest(m, comb.fixed=FALSE)
```



## Last period

```
m <- metagen(imp.LastPeriodIOPChangeMean,
              imp.LastPeriodIOPChangeSEM,
              study.name,
              data=df,
              subset=!is.na(imp.LastPeriodIOPChangeSEM) & MIGsYorN == 'N',
              byvar=subtype)
forest(m, comb.fixed=FALSE)
```

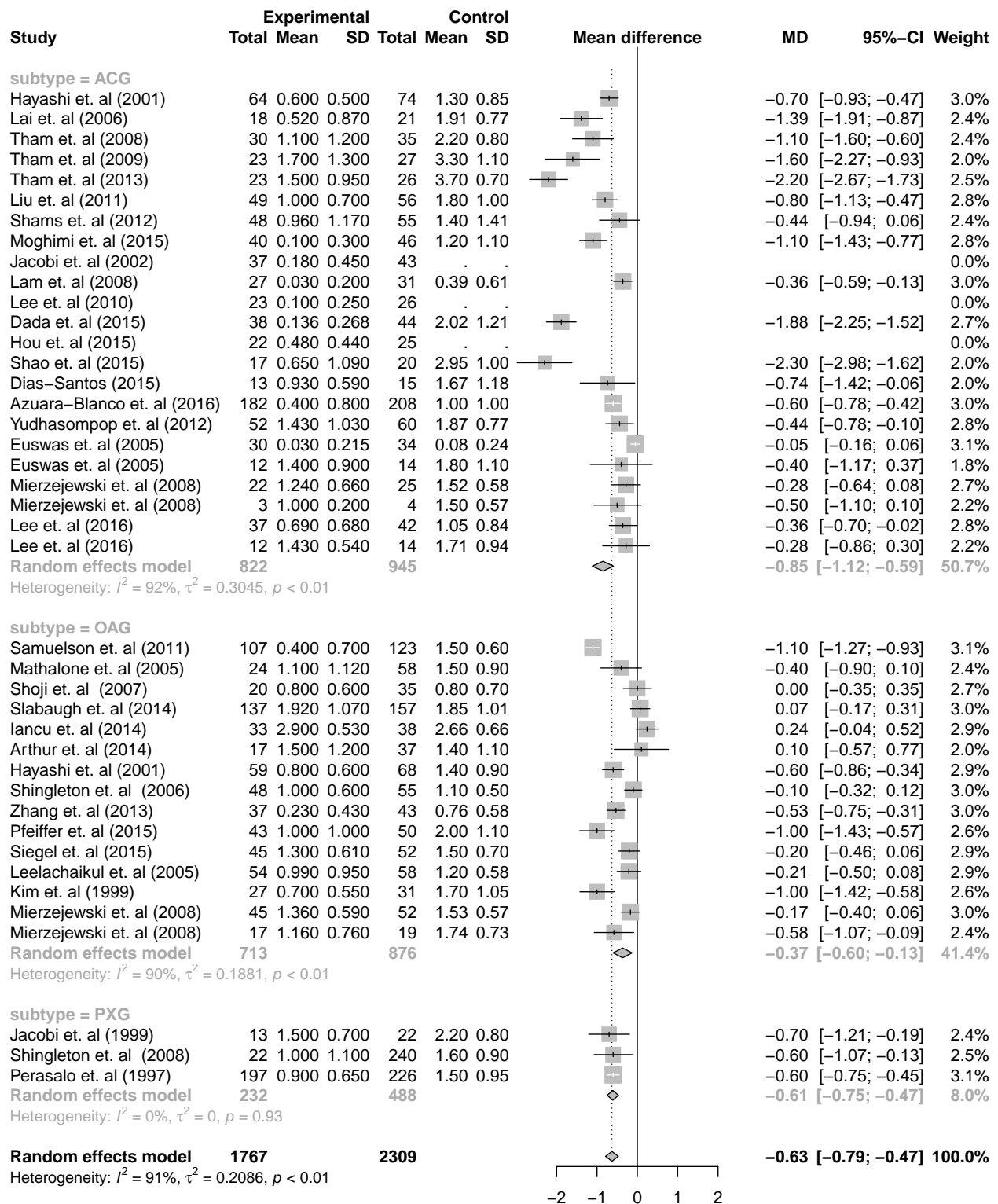




## Meds

```
m <- metacont(imp.LastPeriodEyes,
               RxPostOpMean,
               imp.RxPostOpStdDev,
               PreOpEyes,
```

```
RxPreOpMean,  
imp.RxPreOpStdDev,  
study.name,  
data=df,  
subset=!is.na(imp.RxPostOpStdDev) & !is.na(imp.RxPostOpStdDev) & MIGsYorN == 'N',  
byvar=subtype)  
forest(m, comb.fixed=FALSE)
```



## Correlation between meds and drop in IOP

How is IOP drop related to change in meds? Two hypotheses:

- Those studies that see the largest IOP drops also have drop in meds, as doctors see that can use the

newfound *slack* to decrease the number of meds people take

- The studies that see the largest IOP drops are those that don't change meds, because dropping meds would also increase IOP

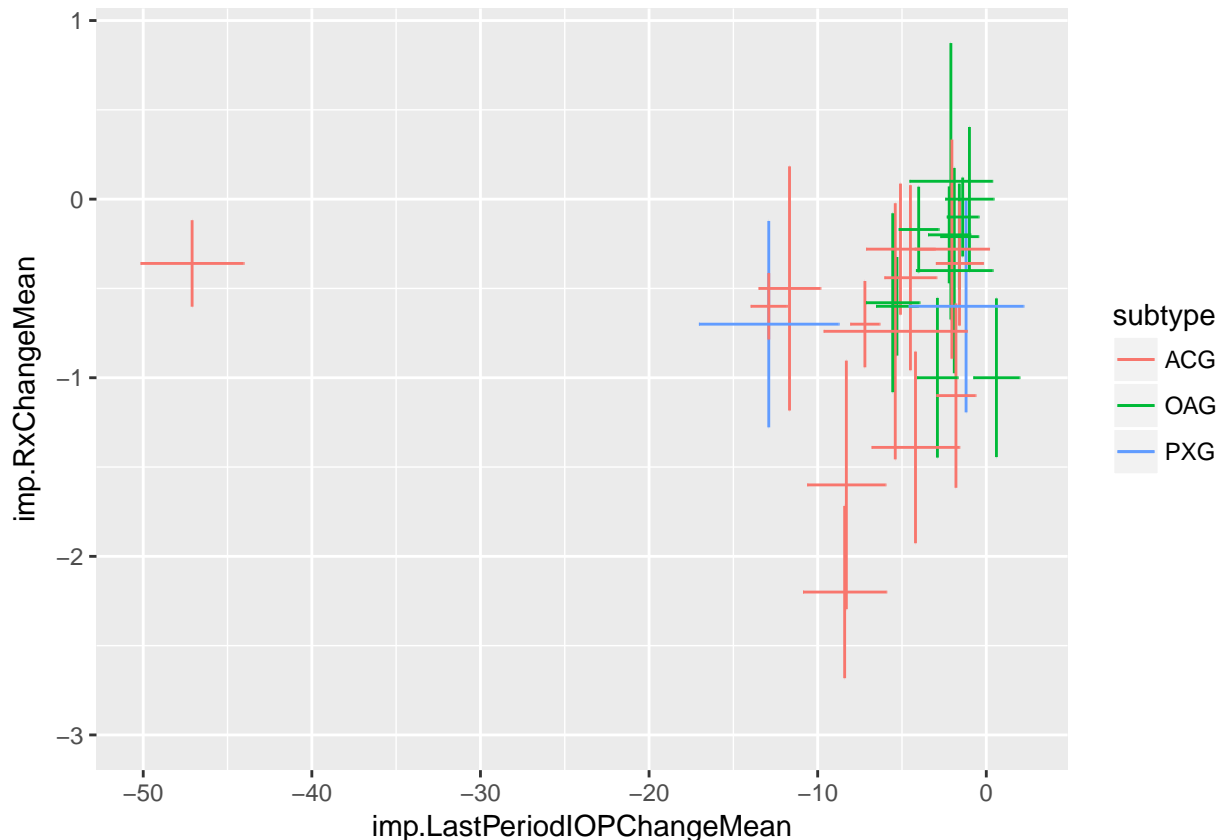
So which is it?

```
df <- df %>% mutate(imp.RxChangeMean = RxPostOpMean - RxPreOpMean,
  imp.RxChangeSEM = sqrt(imp.RxPostOpStdDev ** 2 + imp.RxPreOpStdDev ** 2) / sqrt(imp.LastPeriodIOPChangeSEM),

ggplot(df %>% filter(MIGsYorN == 'N'), aes(x =imp.LastPeriodIOPChangeMean,
  xmin=imp.LastPeriodIOPChangeMean - 1.96*imp.LastPeriodIOPChangeSEM,
  xmax=imp.LastPeriodIOPChangeMean + 1.96*imp.LastPeriodIOPChangeSEM,
  y = imp.RxChangeMean,
  ymin= imp.RxChangeMean - 1.96*imp.RxChangeSEM,
  ymax= imp.RxChangeMean + 1.96*imp.RxChangeSEM,
  color=subtype
)) + geom_errorbar() + geom_errorbarh()
```

## Warning: Removed 22 rows containing missing values (geom\_errorbar).

## Warning: Removed 22 rows containing missing values (geom\_errorbarh).



```
ggplot(df %>% filter(subtype=='OAG' & MIGsYorN == 'N'), aes(x =imp.LastPeriodIOPChangeMean,
  xmin=imp.LastPeriodIOPChangeMean - 1.96*imp.LastPeriodIOPChangeSEM,
  xmax=imp.LastPeriodIOPChangeMean + 1.96*imp.LastPeriodIOPChangeSEM,
  y = imp.RxChangeMean,
  ymin= imp.RxChangeMean - 1.96*imp.RxChangeSEM,
  ymax= imp.RxChangeMean + 1.96*imp.RxChangeSEM,
  label=study.name
```

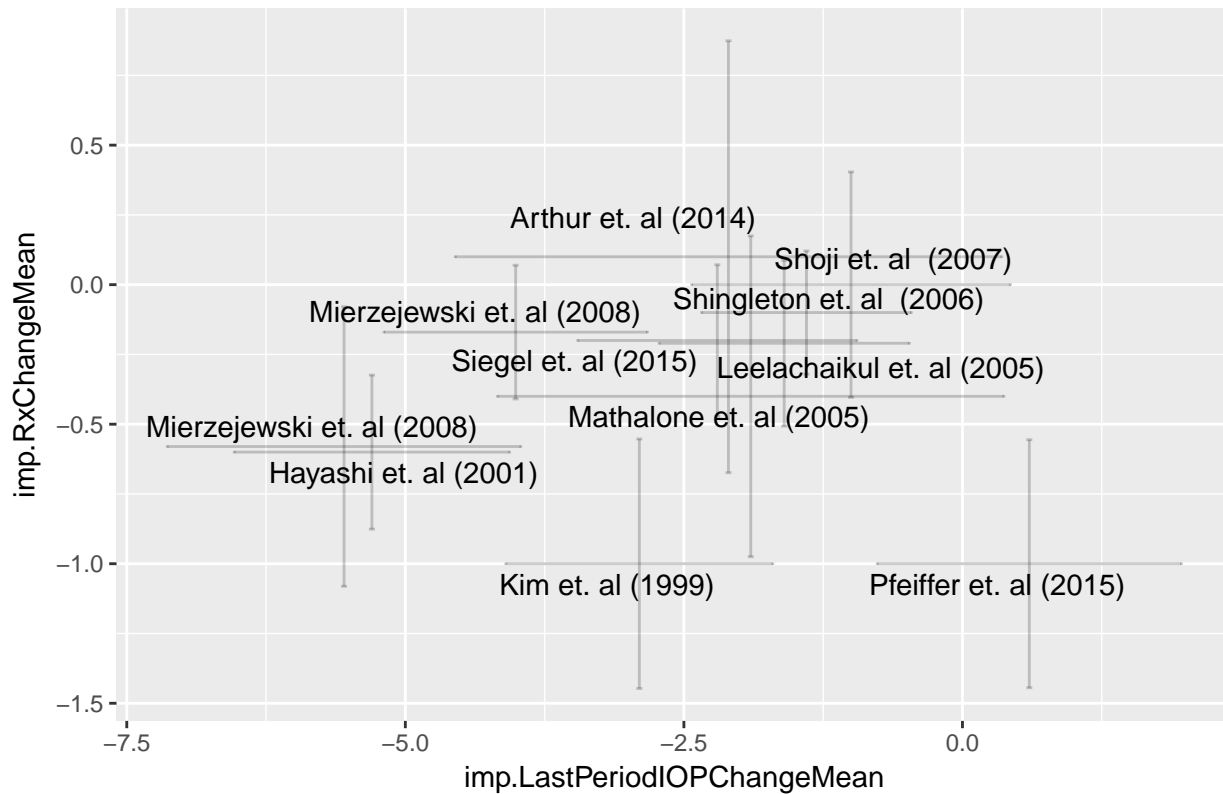
```
)) + geom_errorbar(alpha=.2) + geom_errorbarh(alpha=.2) + ggtitle('OAG only') + geom_text
```

```
## Warning: Removed 7 rows containing missing values (geom_errorbar).
```

```
## Warning: Removed 7 rows containing missing values (geom_errorbarh).
```

```
## Warning: Removed 7 rows containing missing values (geom_text_repel).
```

## OAG only



In fact, apart from the Pfeiffer et al. (2015) study, there is an apparent positive correlation between the two effect sizes: studies with larger drops in IOP also tend to see larger drops in Rx.

```
draw.corr <- function(filter.out) {
  if(filter.out) {
    d_ <- df %>% filter(!(study.name %in% c("Pfeiffer et. al (2015)")))
  } else {
    d_ <- df
  }
  d_ <- d_ %>% filter(subtype=='OAG' & MIGsYorN == 'N') %>%
    dplyr::mutate(x = rnorm(n = n(), mean=imp.LastPeriodIOPChangeMean, sd = imp.LastPeriodIOPChangeSEM),
                  y = rnorm(n = n(), mean=imp.RxChangeMean, sd = imp.RxChangeSEM))
  with(d_ %>% filter(!is.na(x) & !is.na(y)), cor(x, y))
}

cat("Mean +- SE correlation, without Pfeiffer et al\n")
```

```
## Mean +- SE correlation, without Pfeiffer et al
```

```
drawn.corr <- replicate(n = 100, draw.corr(TRUE))
mean(drawn.corr)
```

```
## [1] 0.4125272
sd(drawn.corr)

## [1] 0.204113
cat("Mean +- SE correlation, with Pfeiffer et al\n")

## Mean +- SE correlation, with Pfeiffer et al
drawn.corr <- replicate(n = 100, draw.corr(FALSE))
mean(drawn.corr)

## [1] 0.0485191
sd(drawn.corr)

## [1] 0.1822033
```

## Impute under every scenario

Look at what happens depending on how we impute the data.

```
results <- list()
for(loss in c('lo', 'hi')) {
  for(corr in c('lo', 'hi')) {
    for(delta in c('lo', 'hi')) {
      df <- impute.df(df, losses[[loss]], corrs[[corr]], deltas[[delta]])
      s <- (metagen(imp.SixMoIOPChangeMean,
                    imp.SixMoIOPChangeSEM,
                    study.name,
                    data=df,
                    subset=!is.na(imp.SixMoIOPChangeSEM) & MIGsYorN == 'N',
                    byvar=subtype))
      row <- data.frame(loss=loss, corr=corr, delta=delta, m=s$TE.random.w, ci.lo=s$upper.random.w, ci.hi=s$lower.random.w)
      results[[length(results) + 1]] <- row

      s <- (metagen(imp.OneYIOPChangeMean,
                    imp.OneYIOPChangeSEM,
                    study.name,
                    data=df,
                    subset=!is.na(imp.OneYIOPChangeSEM) & MIGsYorN == 'N',
                    byvar=subtype))
      row <- data.frame(loss=loss, corr=corr, delta=delta, m=s$TE.random.w, ci.lo=s$upper.random.w, ci.hi=s$lower.random.w)
      results[[length(results) + 1]] <- row

      s <- (metagen(imp.LastPeriodIOPChangeMean,
                    imp.LastPeriodIOPChangeSEM,
                    study.name,
                    data=df,
                    subset=!is.na(imp.LastPeriodIOPChangeSEM) & MIGsYorN == 'N',
                    byvar=subtype))
      row <- data.frame(loss=loss, corr=corr, delta=delta, m=s$TE.random.w, ci.lo=s$upper.random.w, ci.hi=s$lower.random.w)
      results[[length(results) + 1]] <- row
    }
  }
}
```

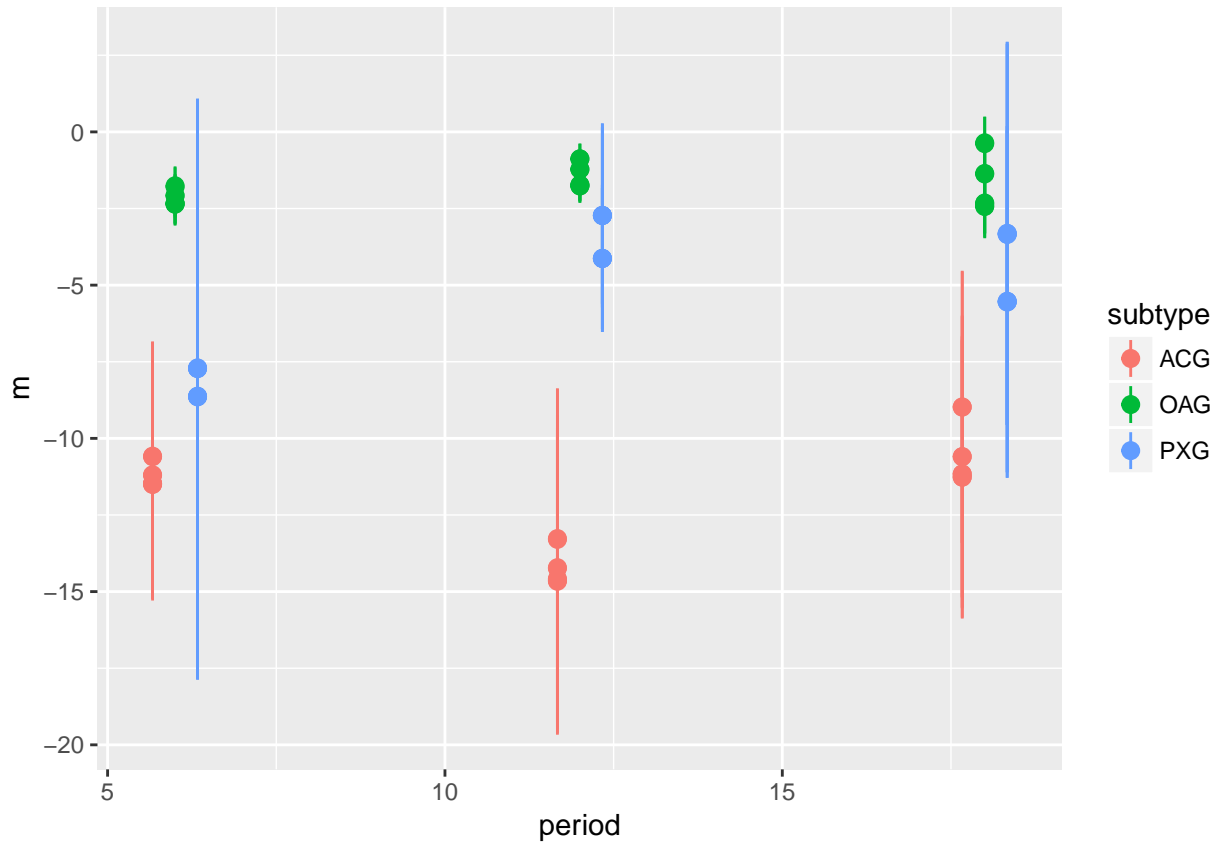
```

}

all.df <- do.call(rbind, results)

p <- position_dodge(width=1)
ggplot(all.df, aes(x=period, y = m, ymin=ci.lo, ymax=ci.hi, color=subtype)) + geom_pointrange(position=p)

```



```

summary(metagen(imp.LastPeriodIOPChangeMean,
  imp.LastPeriodIOPChangeSEM,
  study.name,
  data=df,
  subset=!is.na(imp.LastPeriodIOPChangeSEM) & MIGsYorN == 'N',
  byvar=subtype))

```

```

## Number of studies combined: k = 31
##
##                               95%-CI      z  p-value
## Fixed effect model   -3.6871 [-4.0248; -3.3495] -21.40 < 0.0001
## Random effects model -4.9922 [-7.4405; -2.5439]  -4.00 < 0.0001
##
## Quantifying heterogeneity:
## tau^2 = 46.3999; H = 7.10 [6.55; 7.70]; I^2 = 98.0% [97.7%; 98.3%];
## Rb = 95.9% [93.5%; 98.4%]
##
## Test of heterogeneity:
##      Q d.f.  p-value
## 1512.21  30 < 0.0001

```

```

##
## Results for subgroups (fixed effect model):
##           k           95%-CI      Q    tau^2    I^2    Rb
## subtype = ACG  16 -8.1521 [-8.6777; -7.6265] 967.26  78.74 98.4% 96.6%
## subtype = OAG  12 -0.2848 [-0.7518;  0.1823]  34.76   1.5  68.4% 65.9%
## subtype = PXG   3 -2.6970 [-4.0226; -1.3713]  27.02  27.9  92.6% 91.4%
##
## Test for subgroup differences (fixed effect model):
##           Q d.f.  p-value
## Between groups 483.17    2 < 0.0001
## Within groups 1029.04   28 < 0.0001
##
## Results for subgroups (random effects model):
##           k           95%-CI      Q    tau^2    I^2    Rb
## subtype = ACG  16 -8.9804 [-13.4047; -4.5562] 967.26  78.74 98.4% 96.6%
## subtype = OAG  12 -0.3569 [ -1.2107;  0.4969]  34.76   1.5  68.4% 65.9%
## subtype = PXG   3 -3.3124 [ -9.5628;  2.9381]  27.02  27.9  92.6% 91.4%
##
## Test for subgroup differences (random effects model):
##           Q d.f.  p-value
## Between groups  14.75    2  0.0006
##
## Details on meta-analytical method:
## - Inverse variance method
## - DerSimonian-Laird estimator for tau^2
summary(metagen(imp.LastPeriodIOPChangeMean,
                imp.LastPeriodIOPChangeSEM,
                study.name,
                data=df,
                subset=!is.na(imp.LastPeriodIOPChangeSEM) & MIGsYorN == 'N',
                byvar=subtype))

## Number of studies combined: k = 31
##
##           95%-CI      z  p-value
## Fixed effect model -3.6871 [-4.0248; -3.3495] -21.40 < 0.0001
## Random effects model -4.9922 [-7.4405; -2.5439] -4.00 < 0.0001
##
## Quantifying heterogeneity:
## tau^2 = 46.3999; H = 7.10 [6.55; 7.70]; I^2 = 98.0% [97.7%; 98.3%];
## Rb = 95.9% [93.5%; 98.4%]
##
## Test of heterogeneity:
##           Q d.f.  p-value
## 1512.21    30 < 0.0001
##
## Results for subgroups (fixed effect model):
##           k           95%-CI      Q    tau^2    I^2    Rb
## subtype = ACG  16 -8.1521 [-8.6777; -7.6265] 967.26  78.74 98.4% 96.6%
## subtype = OAG  12 -0.2848 [-0.7518;  0.1823]  34.76   1.5  68.4% 65.9%
## subtype = PXG   3 -2.6970 [-4.0226; -1.3713]  27.02  27.9  92.6% 91.4%
##
## Test for subgroup differences (fixed effect model):
##           Q d.f.  p-value

```



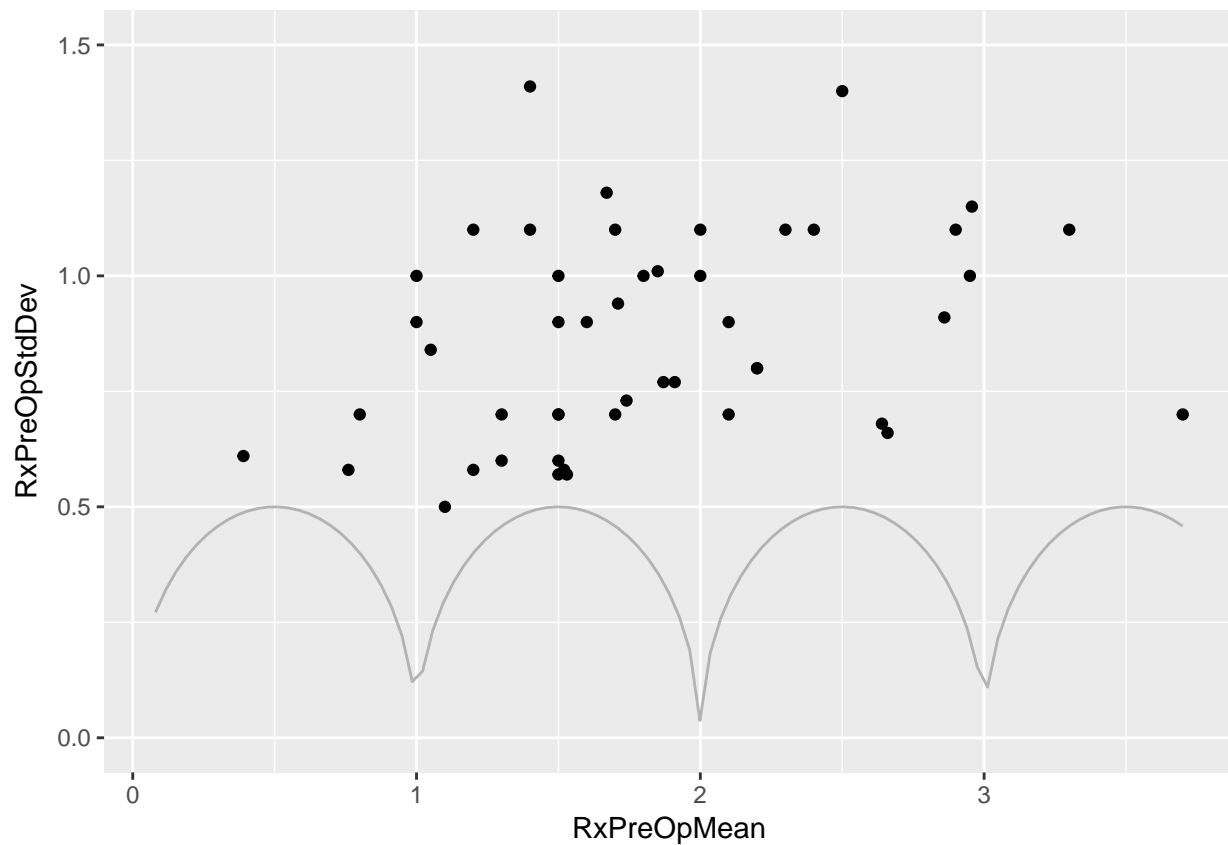
```
## Between groups  483.17    2 < 0.0001
## Within groups  1029.04   28 < 0.0001
##
## Results for subgroups (random effects model):
##           k           95%-CI      Q   tau^2   I^2   Rb
## subtype = ACG  16 -8.9804 [-13.4047; -4.5562] 967.26  78.74 98.4% 96.6%
## subtype = OAG  12 -0.3569 [ -1.2107;  0.4969]  34.76   1.5  68.4% 65.9%
## subtype = PXG   3 -3.3124 [ -9.5628;  2.9381]  27.02  27.9  92.6% 91.4%
##
## Test for subgroup differences (random effects model):
##           Q d.f.  p-value
## Between groups  14.75    2    0.0006
##
## Details on meta-analytical method:
## - Inverse variance method
## - DerSimonian-Laird estimator for tau^2
```

## Sanity check data graphically

Check the relationship between RxPreOpMean and s.d.

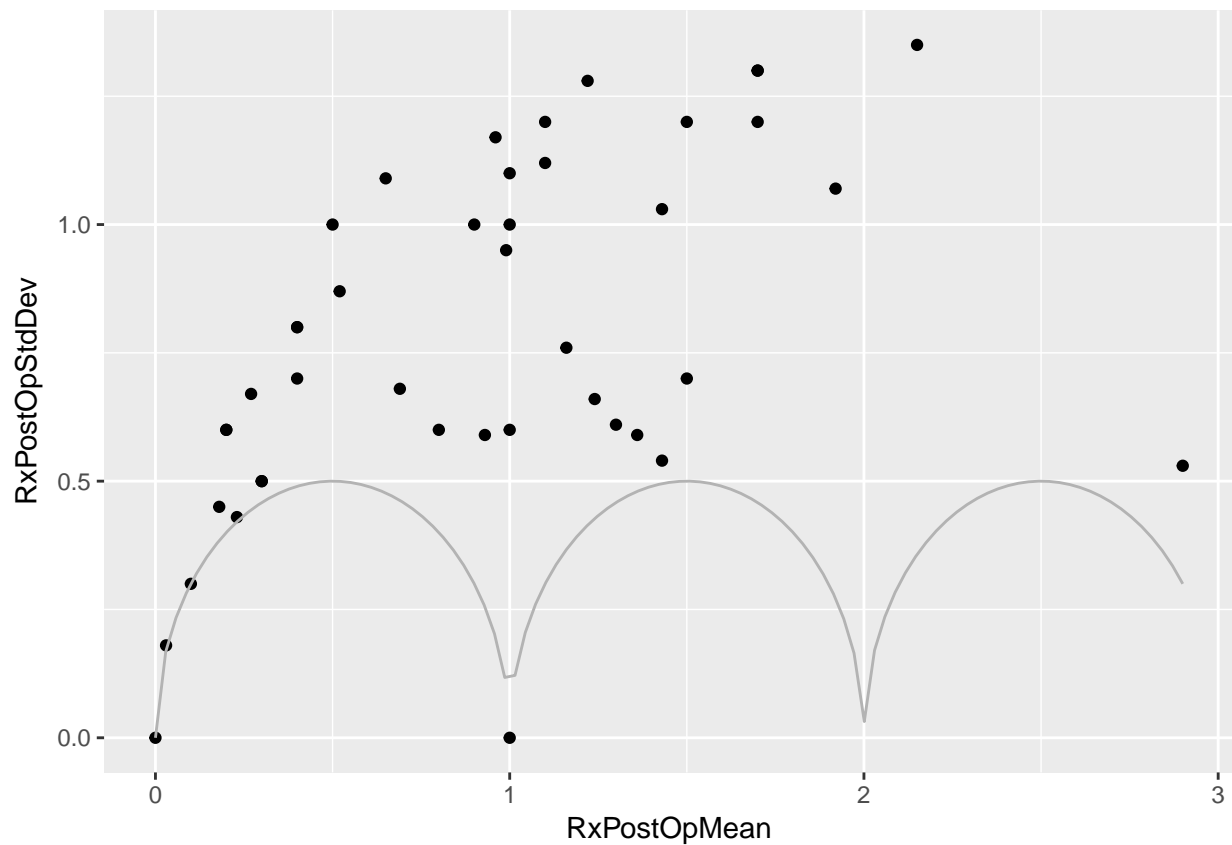
```
ggplot(df, aes(x = RxPreOpMean, y = RxPreOpStdDev)) +
  geom_point() +
  coord_cartesian(y=c(0, 1.5)) +
  stat_function(fun = function(x) sqrt((x - floor(x)) * (1 - (x - floor(x))))), color="gray70")
```

```
## Warning: Removed 19 rows containing missing values (geom_point).
```



```
ggplot(df, aes(x = RxPostOpMean, y = RxPostOpStdDev)) + geom_point() +
  stat_function(fun = function(x) sqrt((x - floor(x)) * (1 - (x - floor(x))))), color="gray70")
```

```
## Warning: Removed 24 rows containing missing values (geom_point).
```

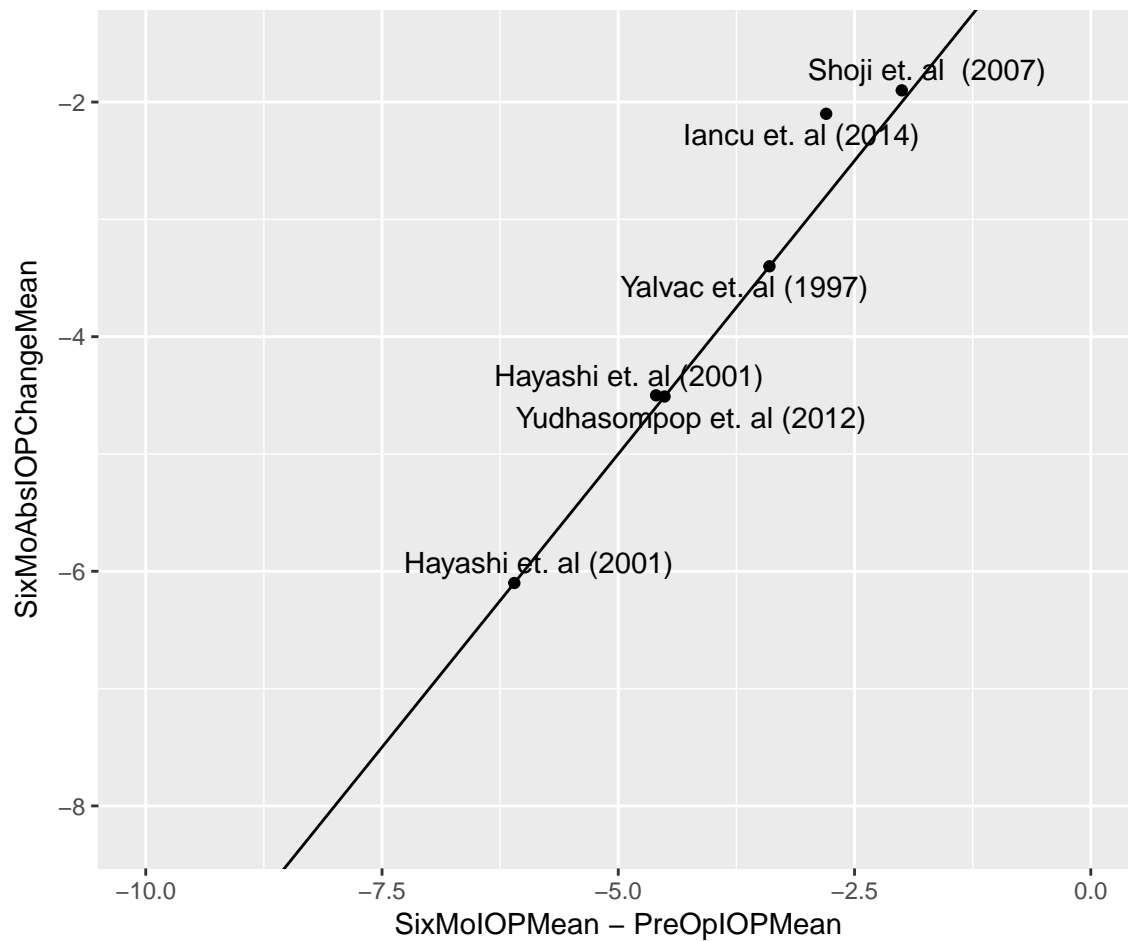


Check that changes add up.

```
ggplot(df, aes(x = SixMoIOPMean - PreOpIOPMean, y = SixMoAbsIOPChangeMean, label = study.name)) +
  geom_point() +
  coord_cartesian(xlim=c(-10, 0)) +
  geom_abline() +
  geom_text_repel()
```

```
## Warning: Removed 61 rows containing missing values (geom_point).
```

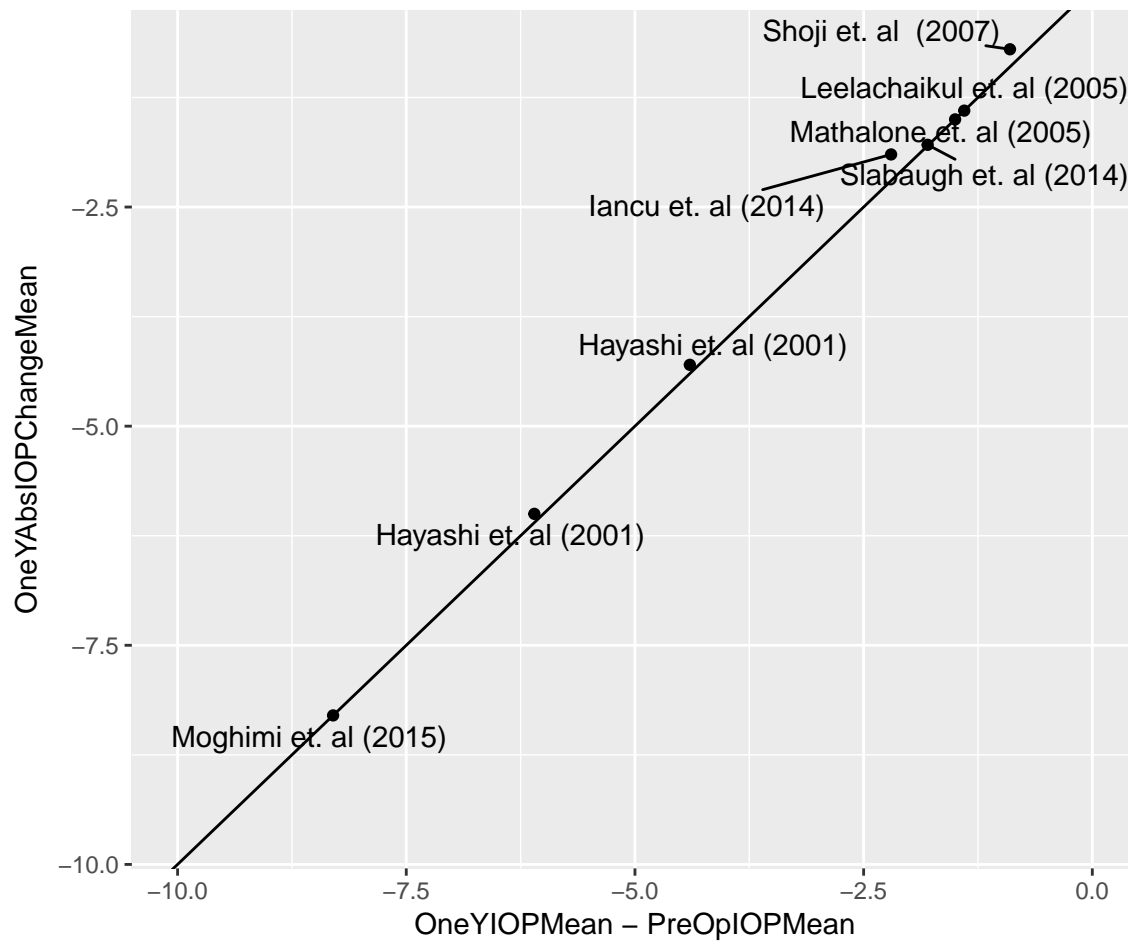
```
## Warning: Removed 61 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x = OneYIOPMean - PreOpIOPMean, y = OneYAbsIOPChangeMean, label = study.name)) +
  geom_point() +
  coord_cartesian(xlim=c(-10, 0)) +
  geom_abline() +
  geom_text_repel()
```

```
## Warning: Removed 59 rows containing missing values (geom_point).
```

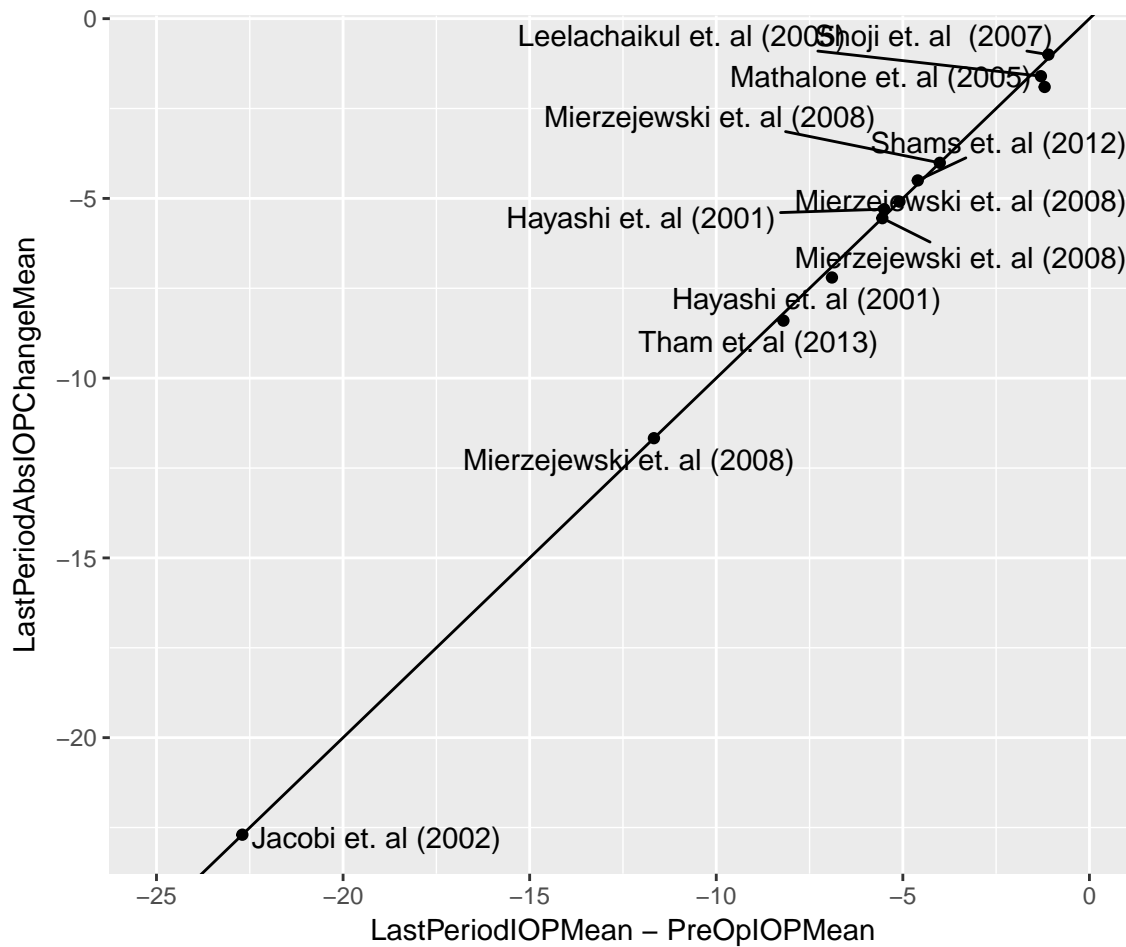
```
## Warning: Removed 59 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x = LastPeriodIOPMean - PreOpIOPMean, y = LastPeriodAbsIOPChangeMean, label = study.name)) +
  geom_point() +
  coord_cartesian(xlim=c(-25, 0)) +
  geom_abline() +
  geom_text_repel()
```

```
## Warning: Removed 55 rows containing missing values (geom_point).
```

```
## Warning: Removed 55 rows containing missing values (geom_text_repel).
```

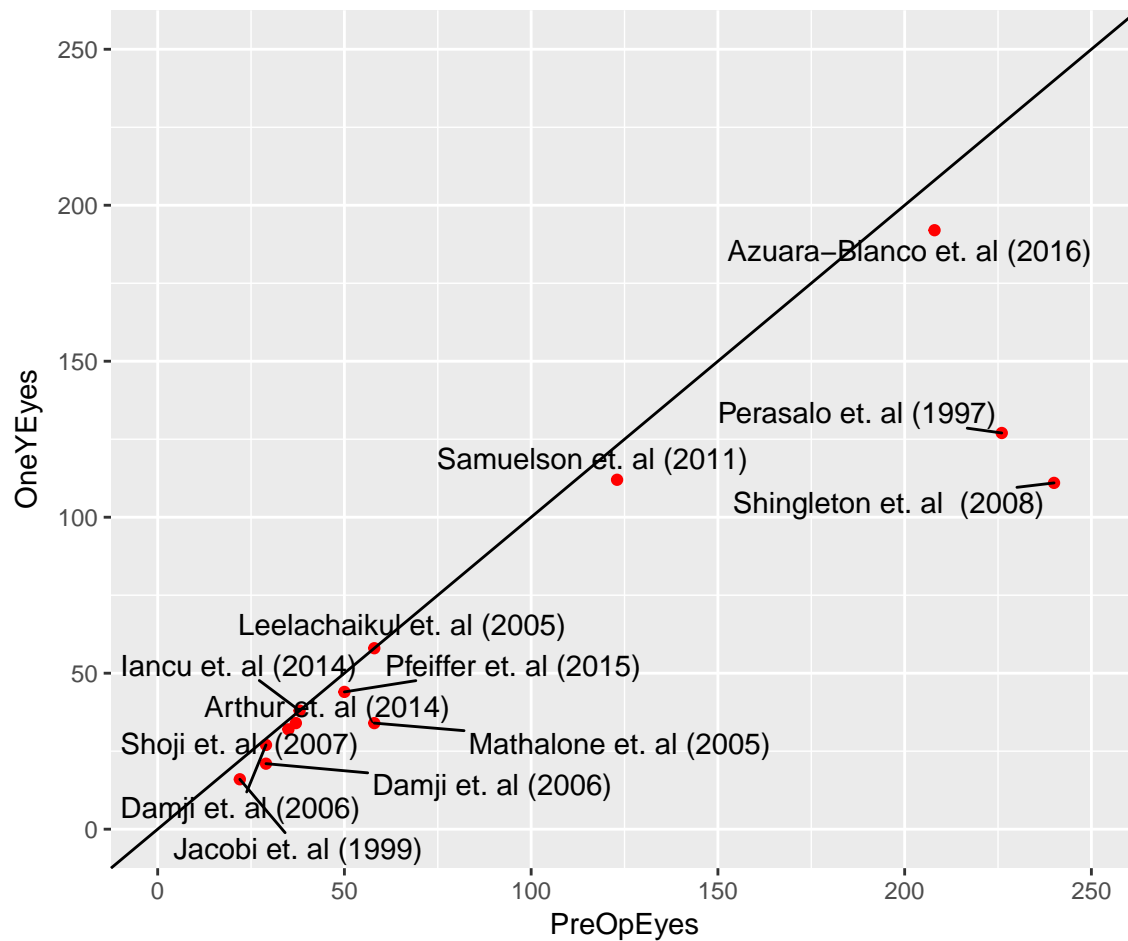


Examine loss at one year.

```
ggplot(df, aes(x=PreOpEyes, y=OneYEyes, label=study.name)) +
  geom_point(color="red") +
  geom_abline() +
  geom_text_repel() + coord_cartesian(xlim=c(0, 250), ylim=c(0, 250))
```

## Warning: Removed 54 rows containing missing values (geom\_point).

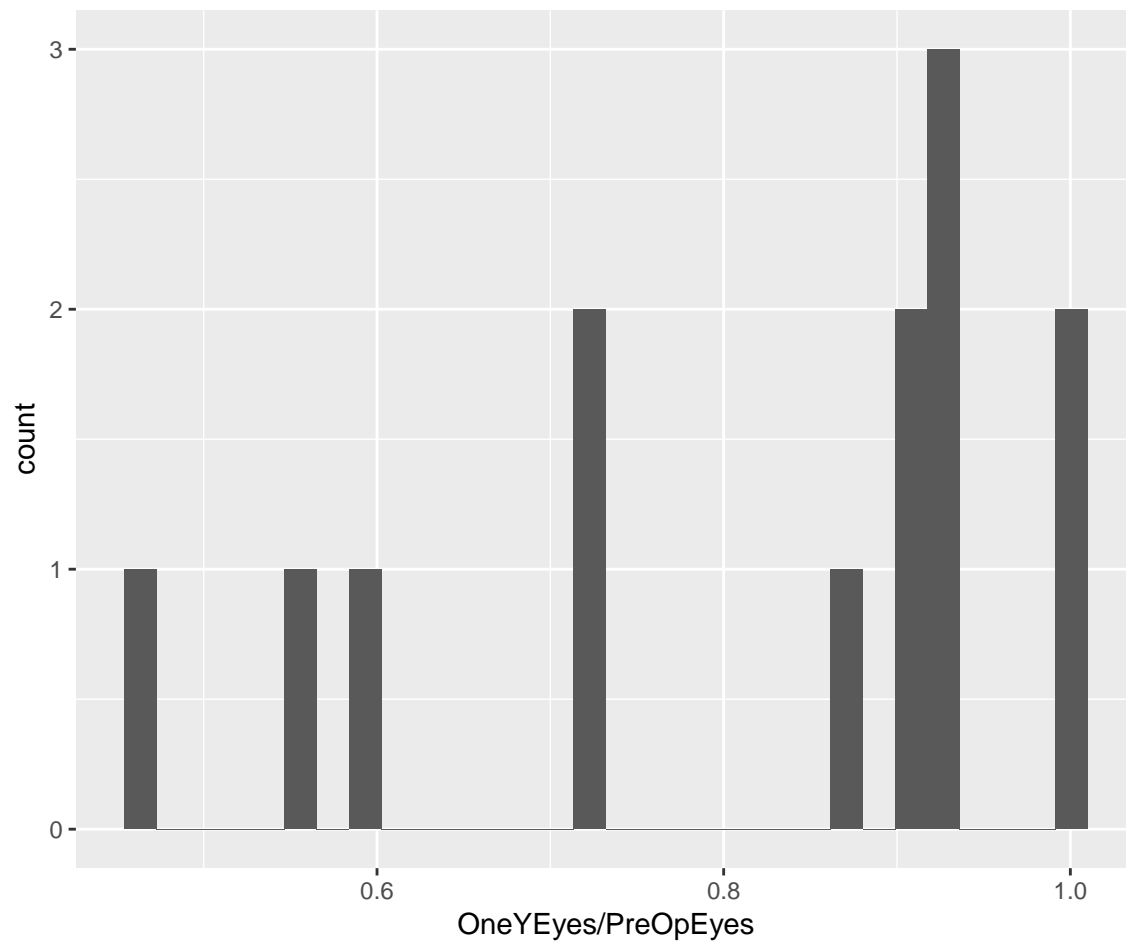
## Warning: Removed 54 rows containing missing values (geom\_text\_repel).



```
ggplot(df, aes(x=OneYEyes / PreOpEyes, label=study.name)) +
  geom_histogram() +
  coord_cartesian()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

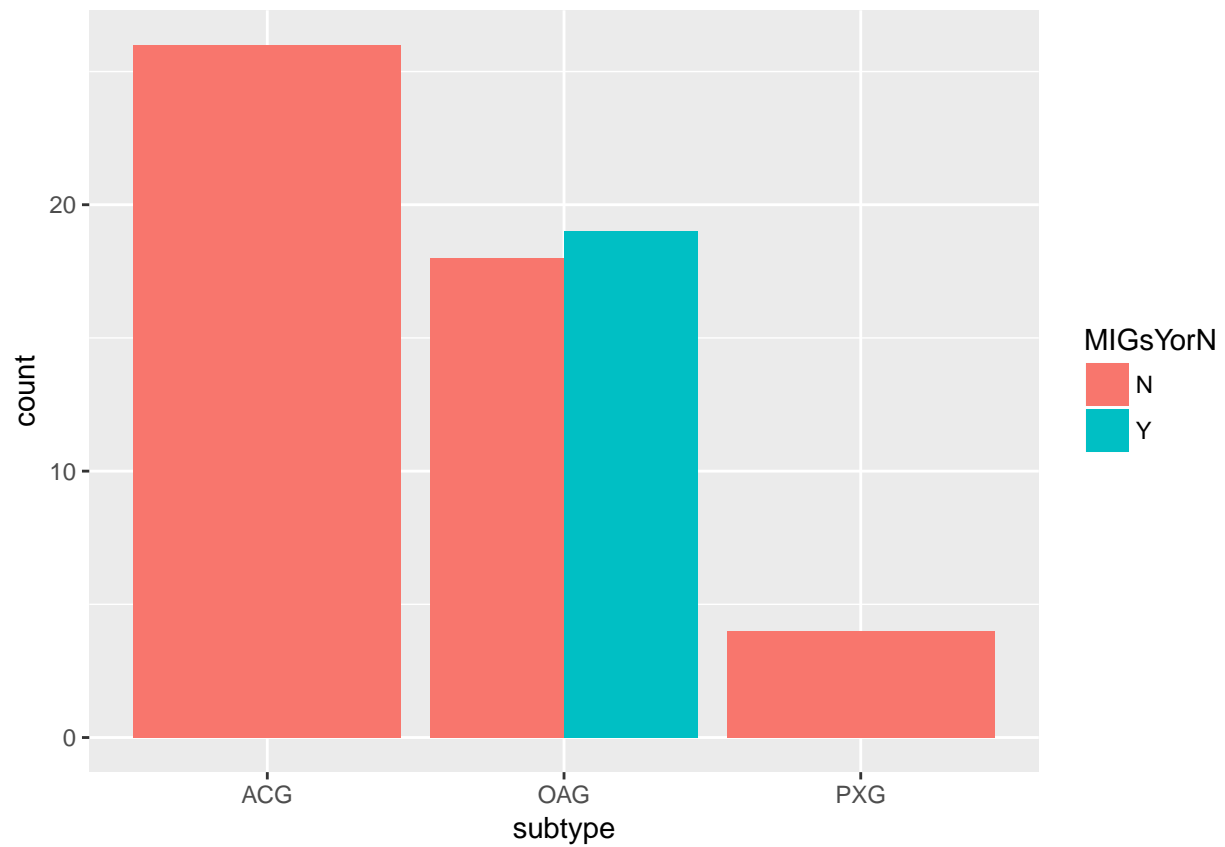
```
## Warning: Removed 54 rows containing non-finite values (stat_bin).
```



```
ggplot(df, aes(x=subtype, fill=MIGsYorN)) + geom_histogram(stat="count", position = 'dodge')
```

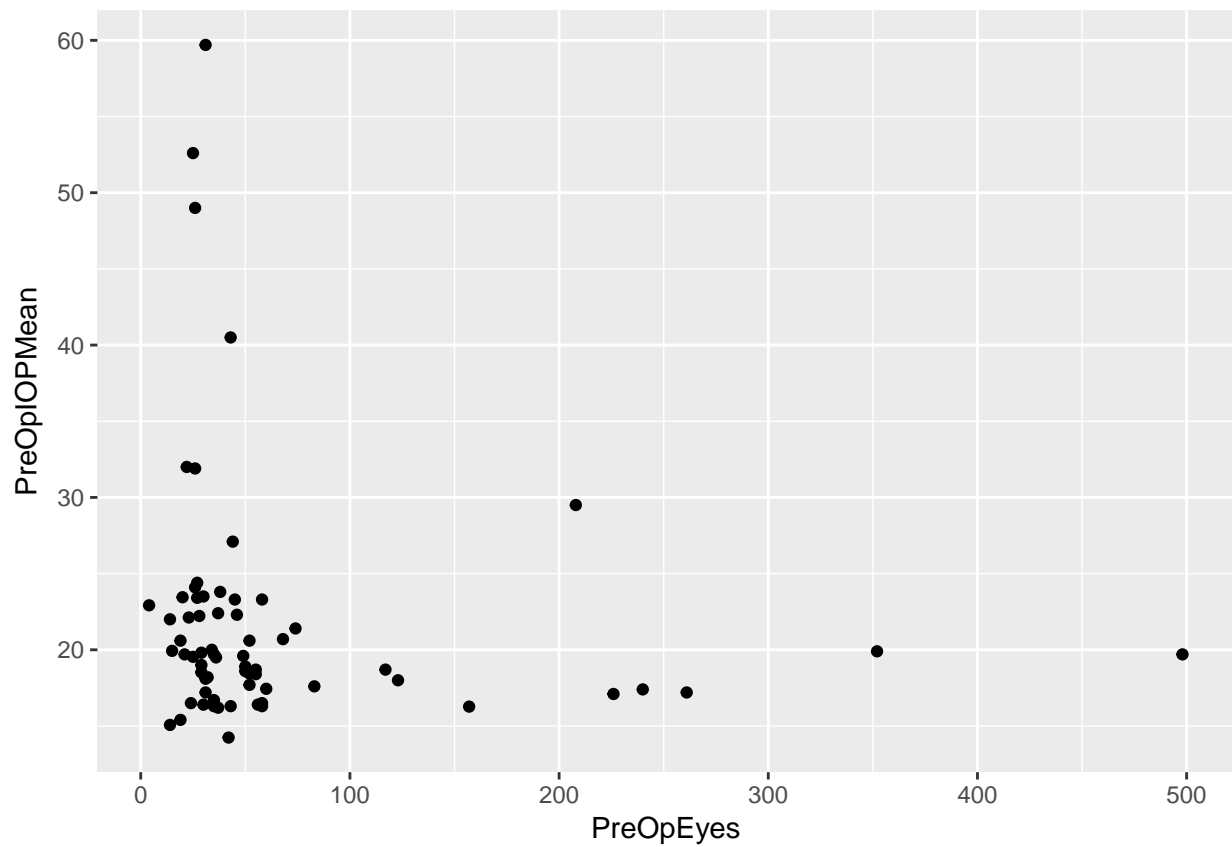
```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```





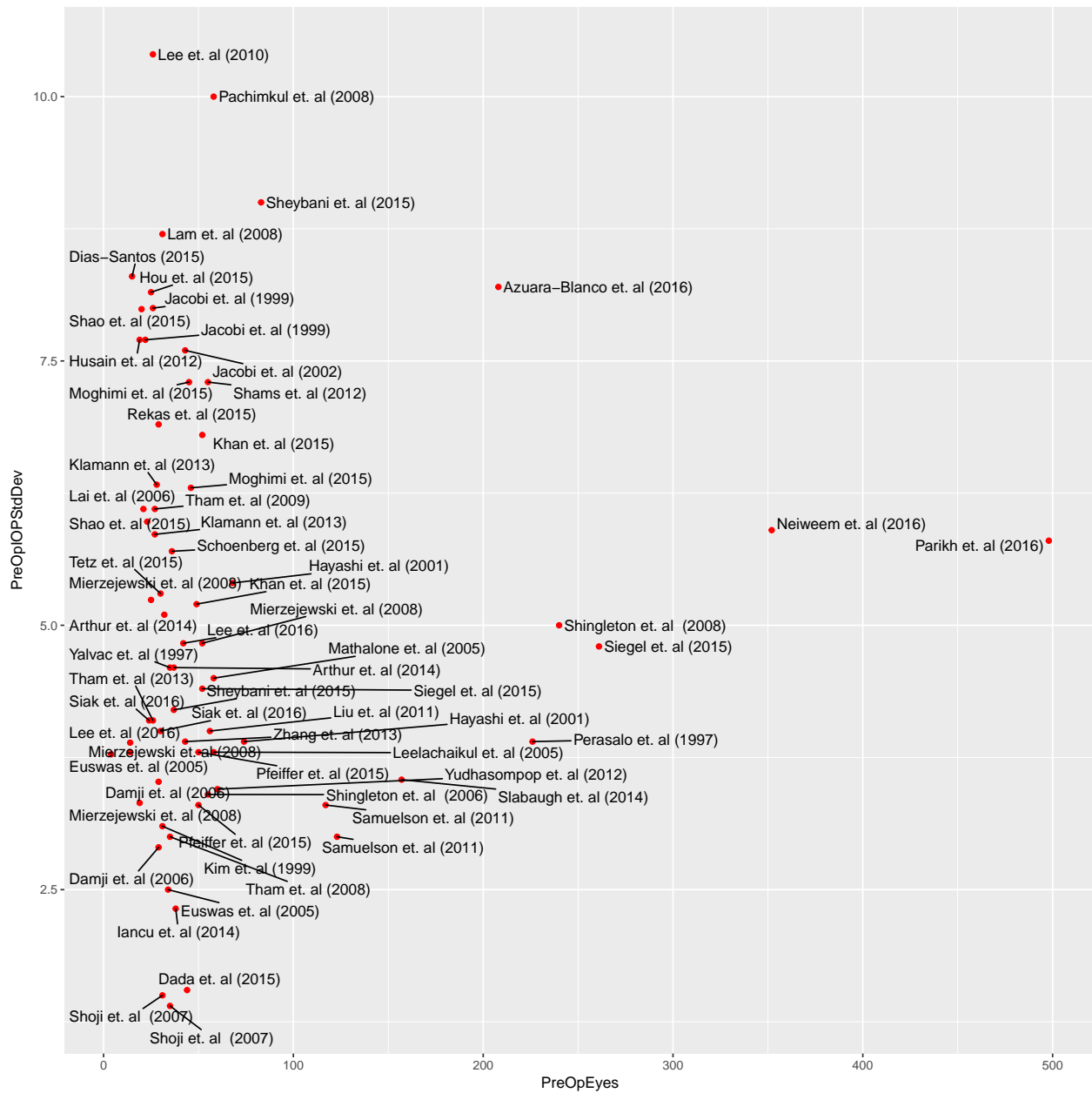
Look at the distribution of eyes and IOP means.

```
ggplot(df, aes(x=PreOpEyes, y=PreOpIOPMean)) + geom_point()
```



Look at number of eyes and standard deviation.

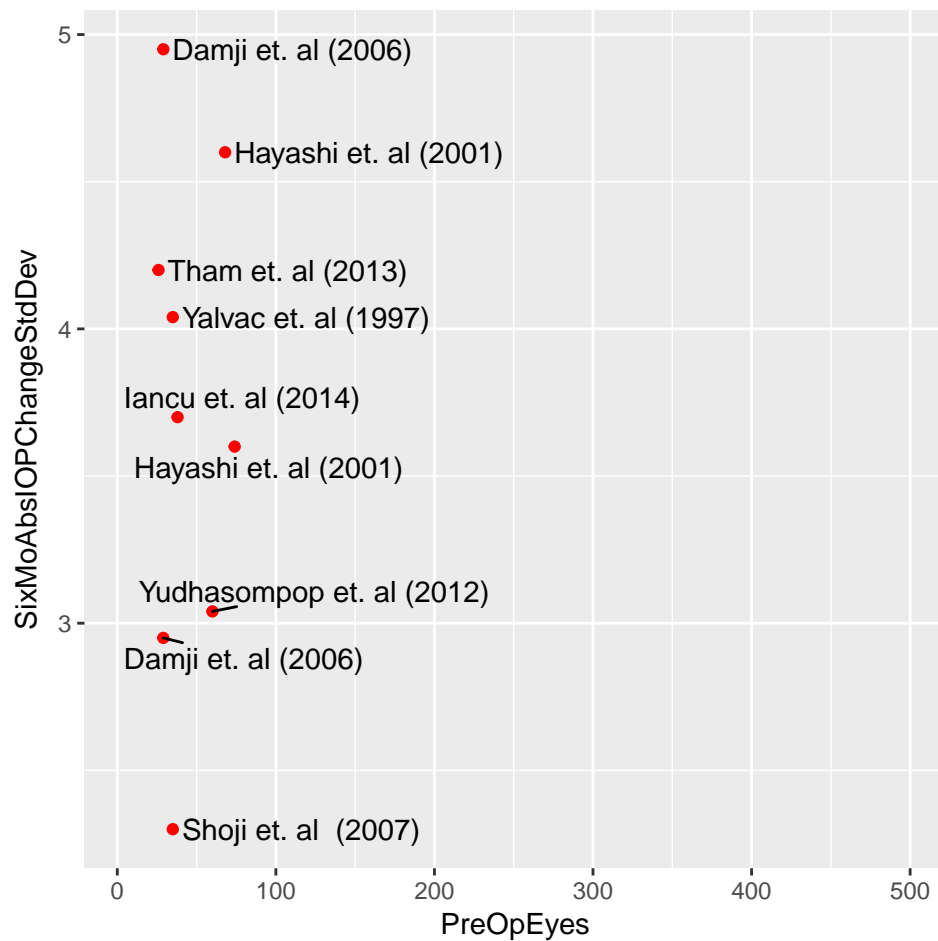
```
ggplot(df, aes(x=PreOpEyes, y=PreOpIOPStdDev, label=study.name)) + geom_point(color="red") + geom_text_
```



```
ggplot(df, aes(x=PreOpEyes, y=SixMoAbsIOPChangeStdDev, label=study.name)) + geom_point(color="red") + g
```

```
## Warning: Removed 58 rows containing missing values (geom_point).
```

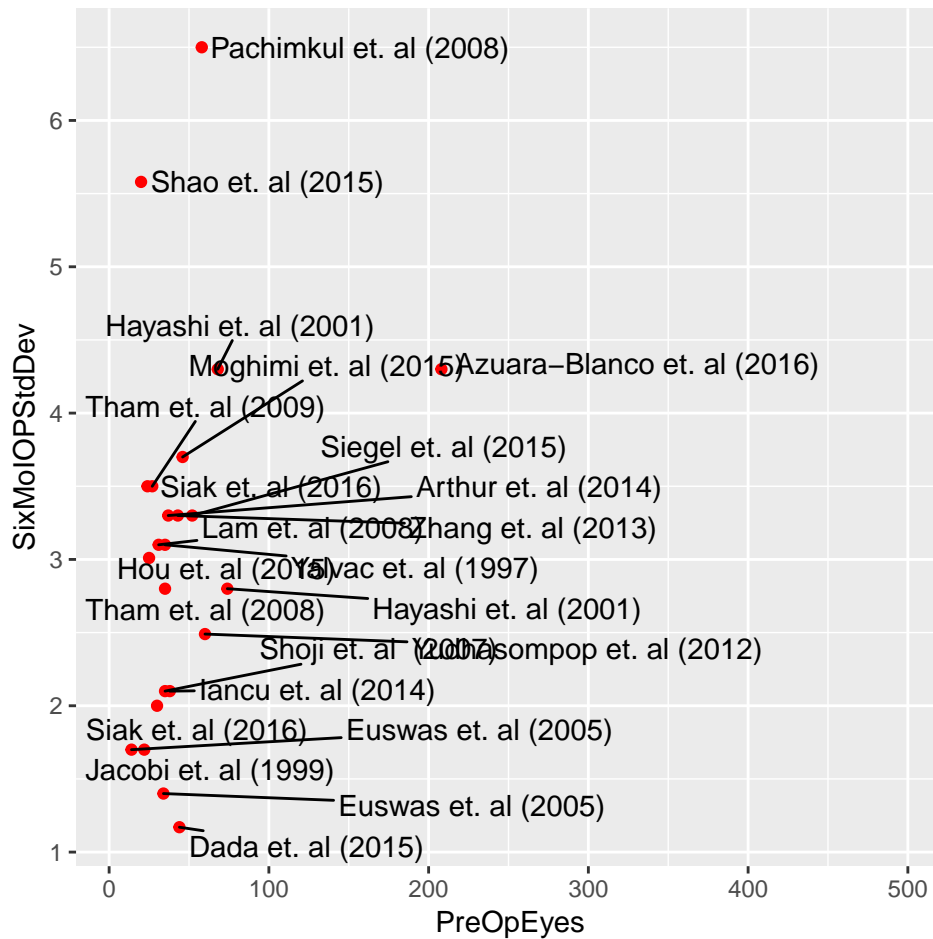
```
## Warning: Removed 58 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x=PreOpEyes, y=SixMoIOPStdDev, label=study.name)) + geom_point(color="red") + geom_text_repel()
```

```
## Warning: Removed 44 rows containing missing values (geom_point).
```

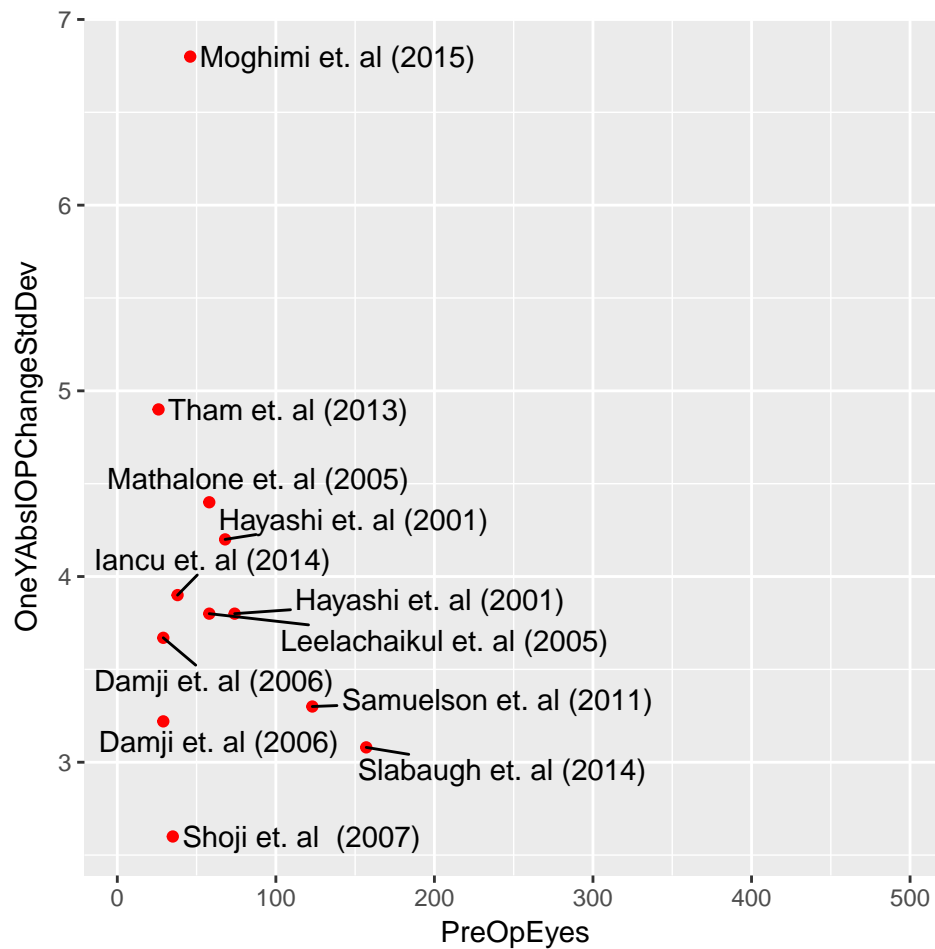
```
## Warning: Removed 44 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x=PreOpEyes, y=OneYAbsIOPChangeStdDev, label=study.name)) + geom_point(color="red") + ge
```

```
## Warning: Removed 55 rows containing missing values (geom_point).
```

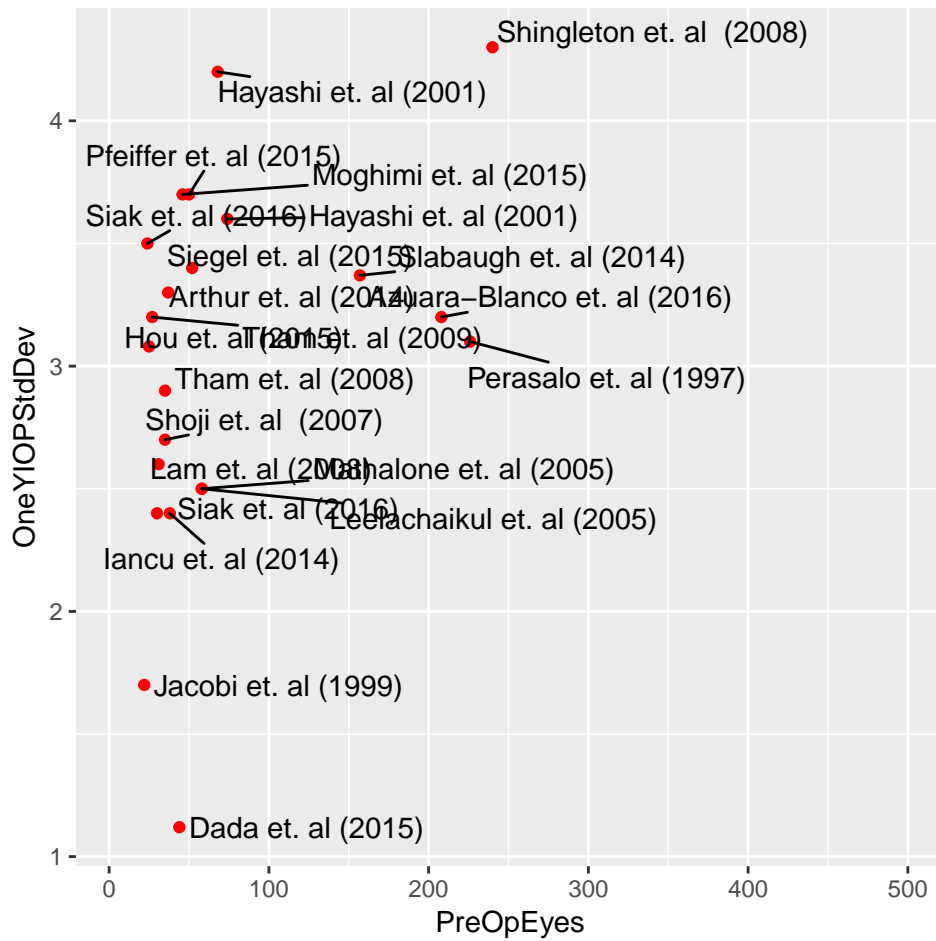
```
## Warning: Removed 55 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x=PreOpEyes, y=OneYIOPStdDev, label=study.name)) + geom_point(color="red") + geom_text_r
```

```
## Warning: Removed 45 rows containing missing values (geom_point).
```

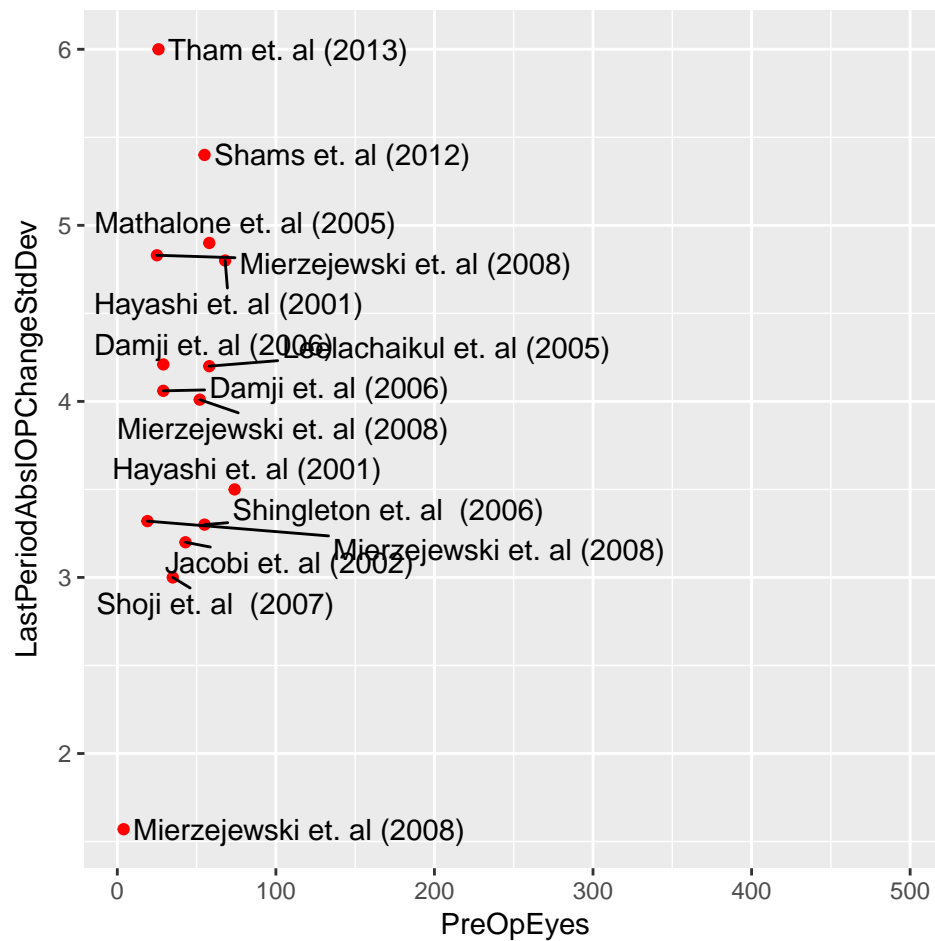
```
## Warning: Removed 45 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x=PreOpEyes, y=LastPeriodAbsIOPChangeStdDev, label=study.name)) + geom_point(color="red",
```

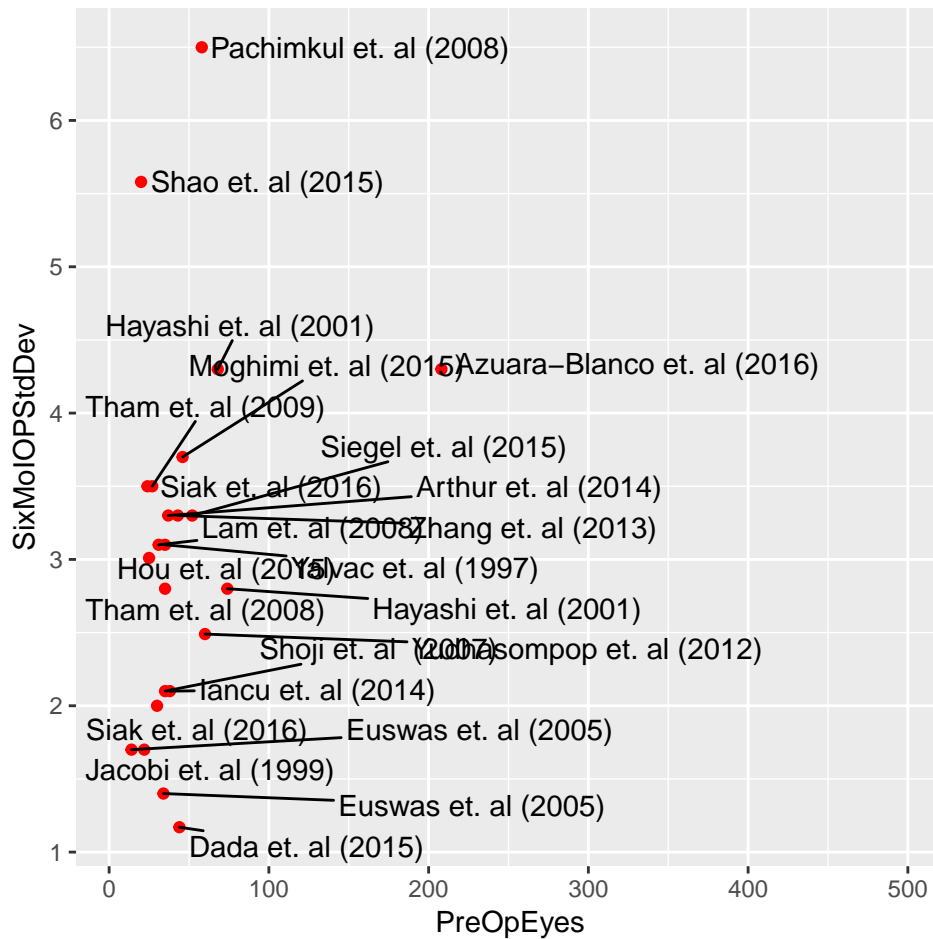
```
## Warning: Removed 52 rows containing missing values (geom_point).
```

```
## Warning: Removed 52 rows containing missing values (geom_text_repel).
```



```
ggplot(df, aes(x=PreOpEyes, y=SixMoIOPStdDev, label=study.name)) + geom_point(color="red") + geom_text_repel(
## Warning: Removed 44 rows containing missing values (geom_point).
## Warning: Removed 44 rows containing missing values (geom_text_repel).
```





## Some notes on the analysis and the studies

- None of the studies are randomized, except the EAGLE one – (2016) Azuara-Bianco et al., Lancet
- There's no control arm in any of the studies
- Main outcome is IOP drop
  - The older studies are phaco + glaucoma surgery
  - The new ones are phaco + MIGS - minimally invasive glaucoma surgery

## Slicings to look at

- MIGS
  - Don't look at MIGS
- Type of glaucoma:
  - OAG -> open angle glaucoma \*\* ~2-3mm \*\*
  - NTG -> normal tension glaucoma ?
  - ACG -> angle closure glaucoma \*\* known to be effective \*\*
  - PXG: pseudo-exfoliation ?

## Dimensions to look at - meta-regression

- Initial severity (IOP before)
- Size of study (number of eyes)
- Year

## Different outcomes

- Primary is IOP drop
  - time points 6 mo, 12 mo, (last time point)
  - most important is 12 months
- Number of meds
  - Huge confound, because it's controlled by the doctor
  - Meds themselves decrease the IOP
  - A handful of studies use washout pre and post (measuring the IOP without meds) to undo the confounding
    - \* EAGLE, Samuelson studies have washout
    - \* Lack of washout will have a tendency to decrease the apparent effectiveness of the studies
  - One med  $\sim$  20% decrease in IOP
  - One med  $:=$  decrease in quality of life
  - RxPostOpMean is at the same time as LastPeriod
- (visual acuity but it's kind of obvious)

## Additional analyses to perform

- Funnel plot for small / medium large studies
- Deal appropriately with multiple arms of same study, e.g. Damji et al., Merz...
- Deal with three forms of lossiness:
  - Absolutes reported, relatives needed
  - Can patch up using estimate of  $\rho - \sqrt{s_1^2 + s_2^2 - 2 * \rho * s_1 * s_2}$
  - Try  $\rho = 0$ ,  $\rho = 0.5$
  - Loss of follow-up
  - Can deal with by assuming that follow up is either MCAR or worse than MCAR
  - Try  $\text{mean\_delta} = 0$ ,  $\text{mean\_delta} = -3$ ,  $\text{mean\_delta} = -5$
  - Not all metrics reported for every study
  - Use mvmeta