par(mfrow=c(2,2), mar = c(4,4,3,1))
n=10000 # ideally 1000000 but plotting may crash your R
n.plotted = 4000
# by rounding to 1 decimal place induces bins w=0.1 wide
for(sigma in c(1,0.1)) {
  true.y=rnorm(n,0,sigma)
  rounded.y = round(true.y,1)
  print(c(var(true.y), var(rounded.y), (0.1)^2 / 12))
  var(rounded.y) - var(true.y), ( (0.1)^2 ) / 12)
  rounding.error = rounded.y - true.y
  txt = paste("Variances: rounded",
  format(round(var(rounded.y),5),nsmall=5),
  "TRUE",format(round(var(true.y),5),nsmall=5),
  "error", format(round(var(rounding.error),
  5),nsmall=5),
  ": Corr'ln[TRUE,error]",
  format(round(cor(true.y,rounding.error),3),nsmall=3)
  )
  plot(true.y[1:n.plotted], rounding.error[1:n.plotted],
  cex=0.1,pch=19,
  main=txt,cex.main=0.8, ylim=c(-0.05,0.05))
  txt = paste("Variances: rounded",
  format(round(var(rounded.y),6)),
  "TRUE",format(round(var(true.y),5),nsmall=5),
  "error", format(round(var(rounding.error),
  5),nsmall=5),
  ": Corr'ln[rounded,error]",
  format(round(cor(rounded.y,rounding.error),
  3),nsmall=3)
  )
  plot(rounded.y[1:n.plotted],
  rounding.error[1:n.plotted],cex=0.1,pch=19,
  main=txt,cex.main=0.8, ylim=c(-0.05,0.05))
  }

# TAKE-HOME MESSAGES
# if MANY bins [top], **********
# errors and true y close to uncorrelated
# variance(errors) << variance(true) & variance(rounded)
# so corrections are minor even if ignore the correlation
# var(rounded) = var(true+error) is close to
# var(true) + var(error), so subtract var(error)
# from var(rounded) to get var(true)
# and use var(error) = (width of bin)^2 / 12
# note that with w=0.1, var(Uniform) = 0.01/12 = 0.00083
# if FEW bins [bottom], ***************
# variance(errors) closer in magnitude to
# variance(true) & variance(rounded)
# so corrections matter more
# errors are now substantially correlated with true y
# so, ignoring this is more crucial, and
# Sheppard’s correction is less accurate.