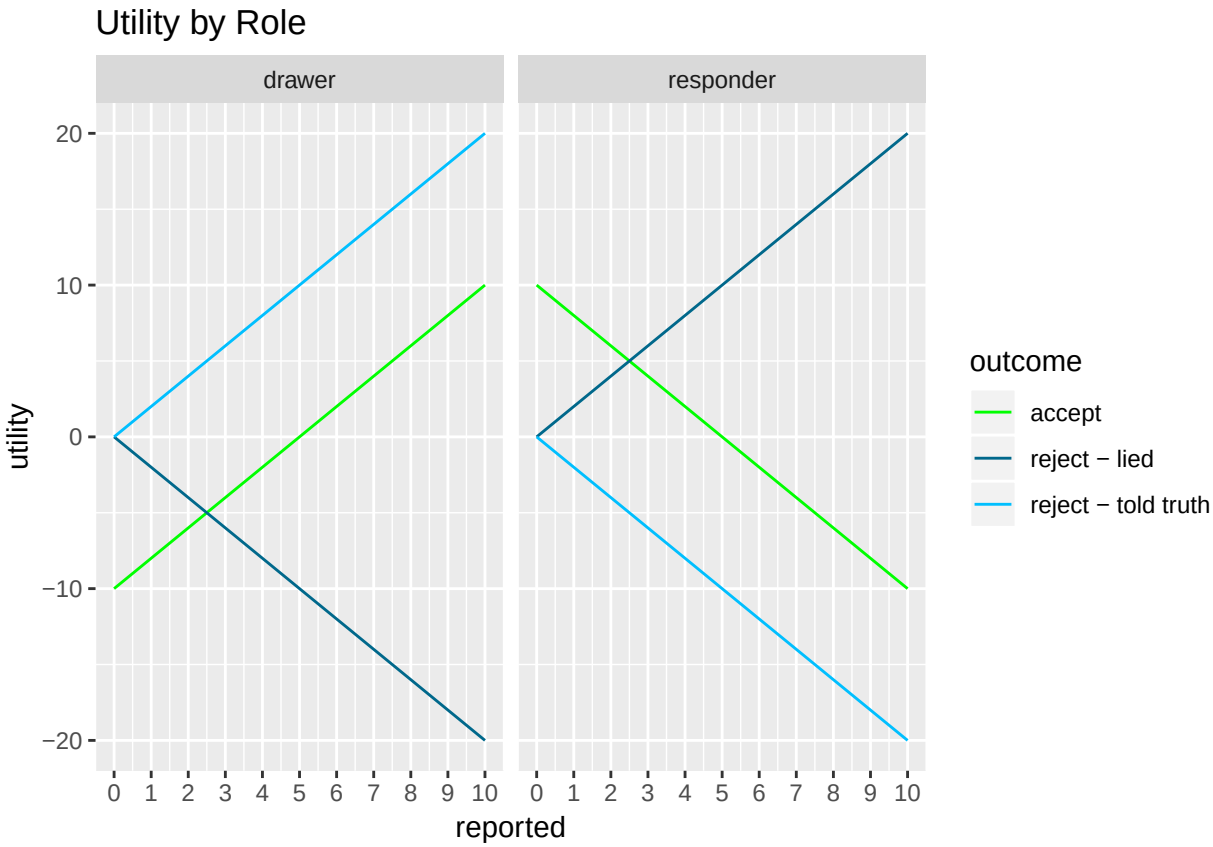
Bullshitter AI

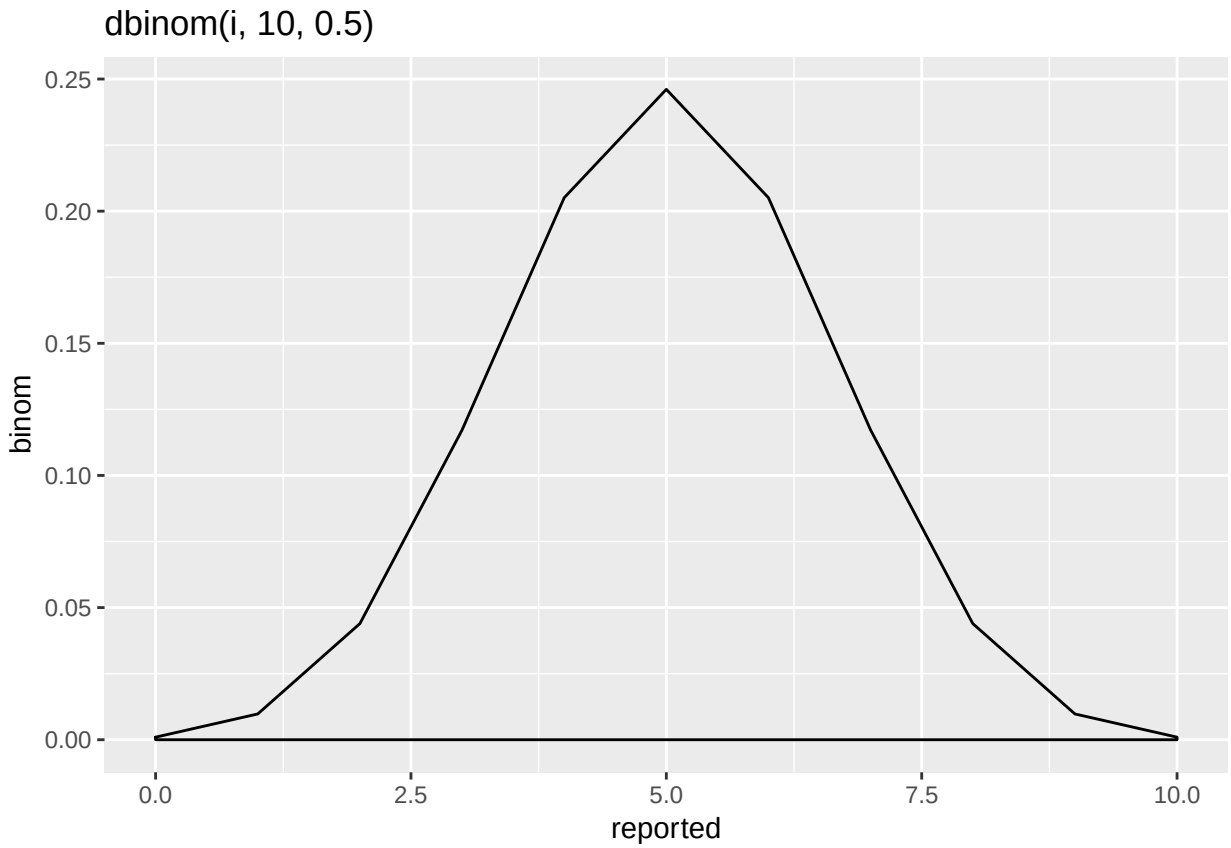
Lauren Oey

11/29/2018

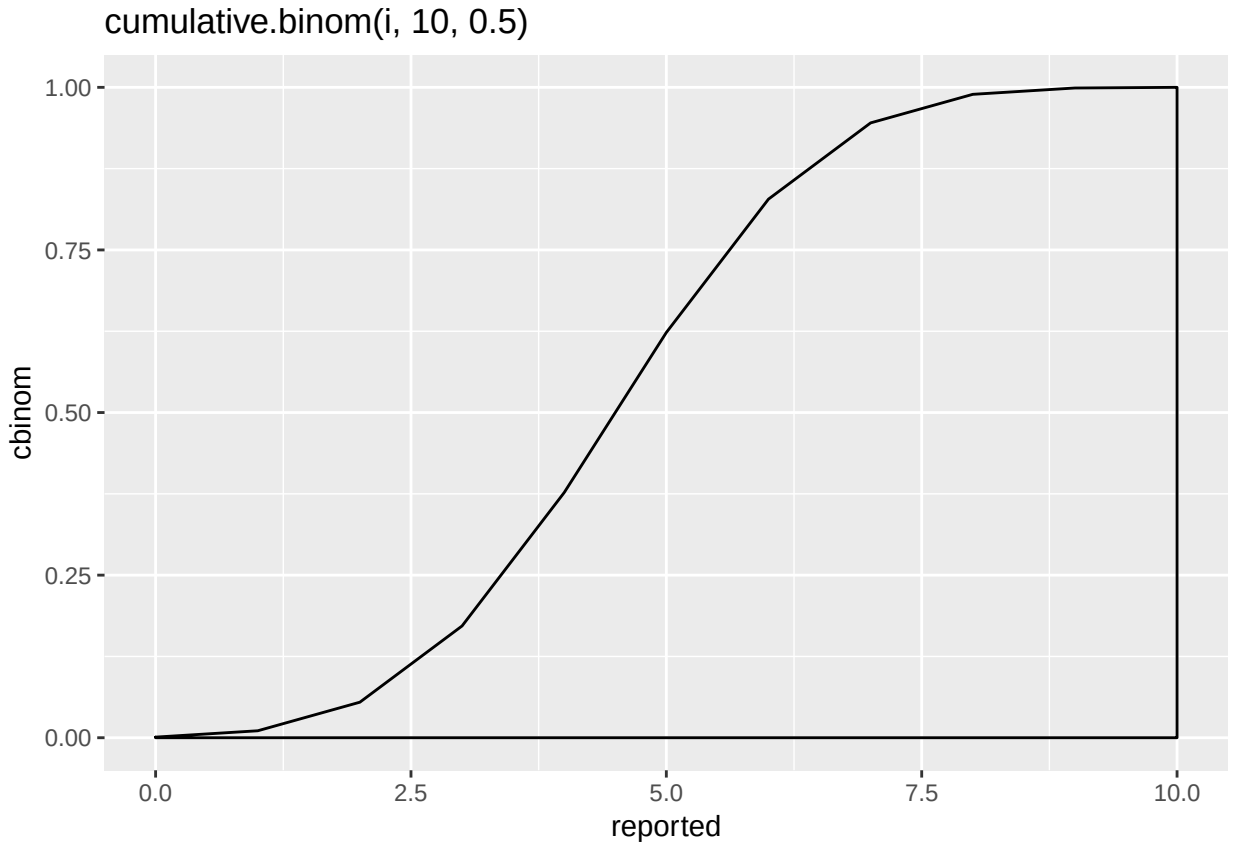
```
utils %>%
  gather("outcome", "utility", 3:5) %>%
  ggplot(aes(x=reported, y=utility, colour=outcome)) +
  geom_line(stat="identity") +
  scale_x_continuous(breaks=pretty(reported, n=10)) +
  scale_colour_manual(labels=c("accept", "reject - lied", "reject - told truth"), values=c("green", "de
  ggtitle("Utility by Role") +
  facet_wrap(~ role)
```




```
probs$binom <- dbinom(probs$reported, 10, 0.5)
probs %>%
  select("reported","binom") %>%
  unique() %>%
  ggplot(aes(x=reported, y=binom)) +
  geom_density(stat="identity") +
  ggtitle("dbinom(i, 10, 0.5)")
```

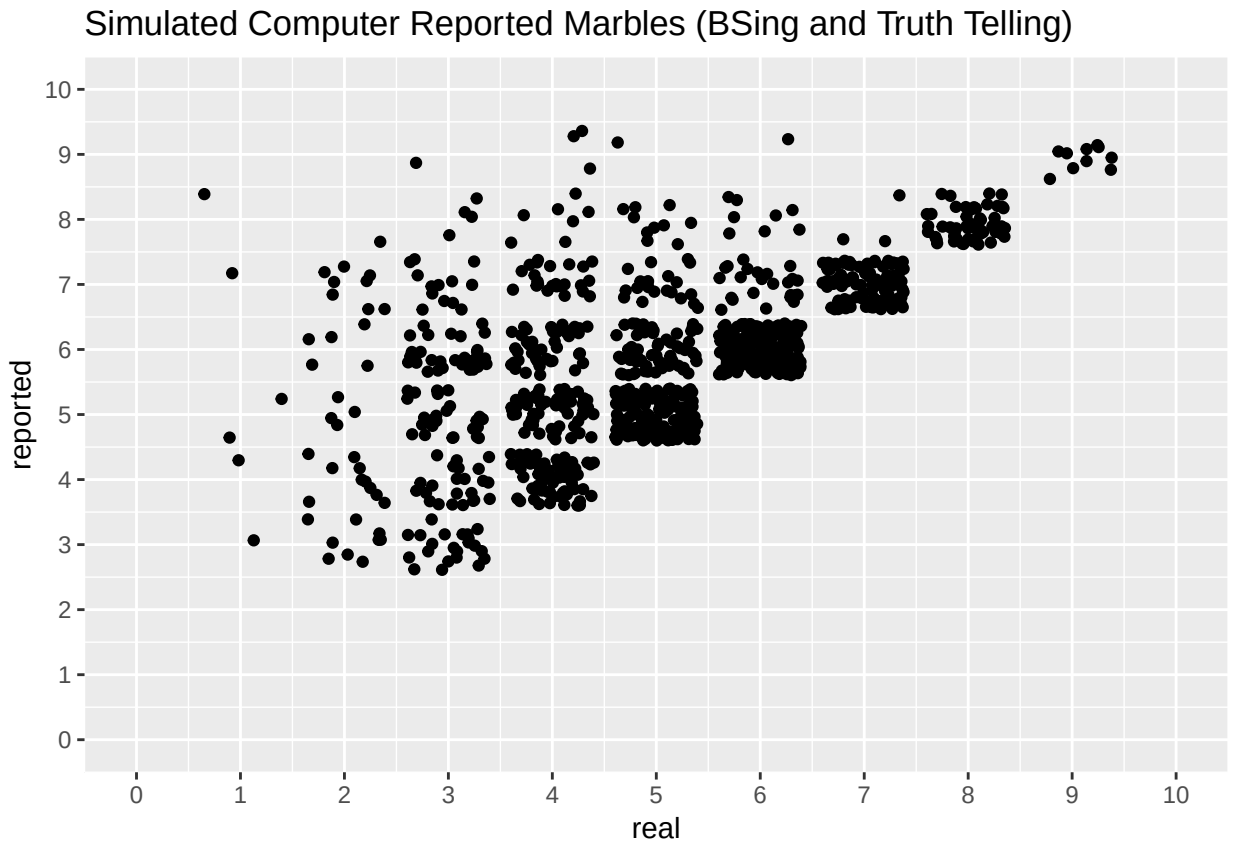



```
probs$cbinom <- pbinom(probs$reported, 10, 0.5)
probs %>%
  select("reported","cbinom") %>%
  unique() %>%
  ggplot(aes(x=reported, y=cbinom)) +
  geom_density(stat="identity") +
  ggtitle("cumulative.binom(i, 10, 0.5)")
```

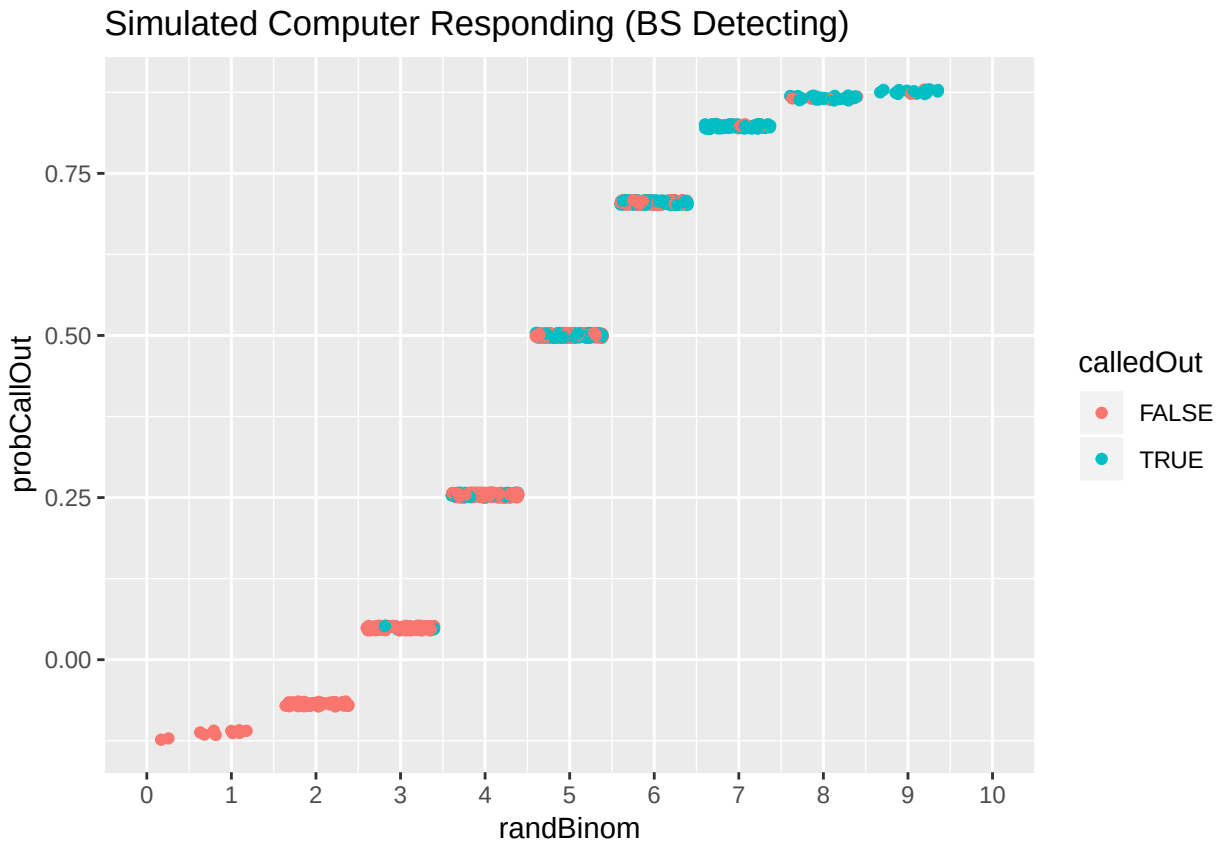



```
responderProbs <- left_join(probs, drawer, by="reported")
rand <- runif(1000)
responderProbsBS.sim <- data.frame(rand, getK = qbinom(rand, 10, 0.5), real=rbinom(1000, 10, 0.5))
#responderProbsBS.sim <- data.frame(rand, getK = rbinom(1000, 10, 0.5), real=rbinom(1000, 10, 0.5))

responderProbsBS.sim %>%
  mutate(lie = getK > real,
         reported = ifelse(lie, getK, real)) %>%
  ggplot(aes(x=real, y=reported)) +
  geom_jitter() +
  scale_x_continuous(limits=c(0,10), breaks=0:10) +
  scale_y_continuous(limits=c(0,10), breaks=0:10) +
  ggtitle("Simulated Computer Reported Marbles (BSing and Truth Telling)")
```

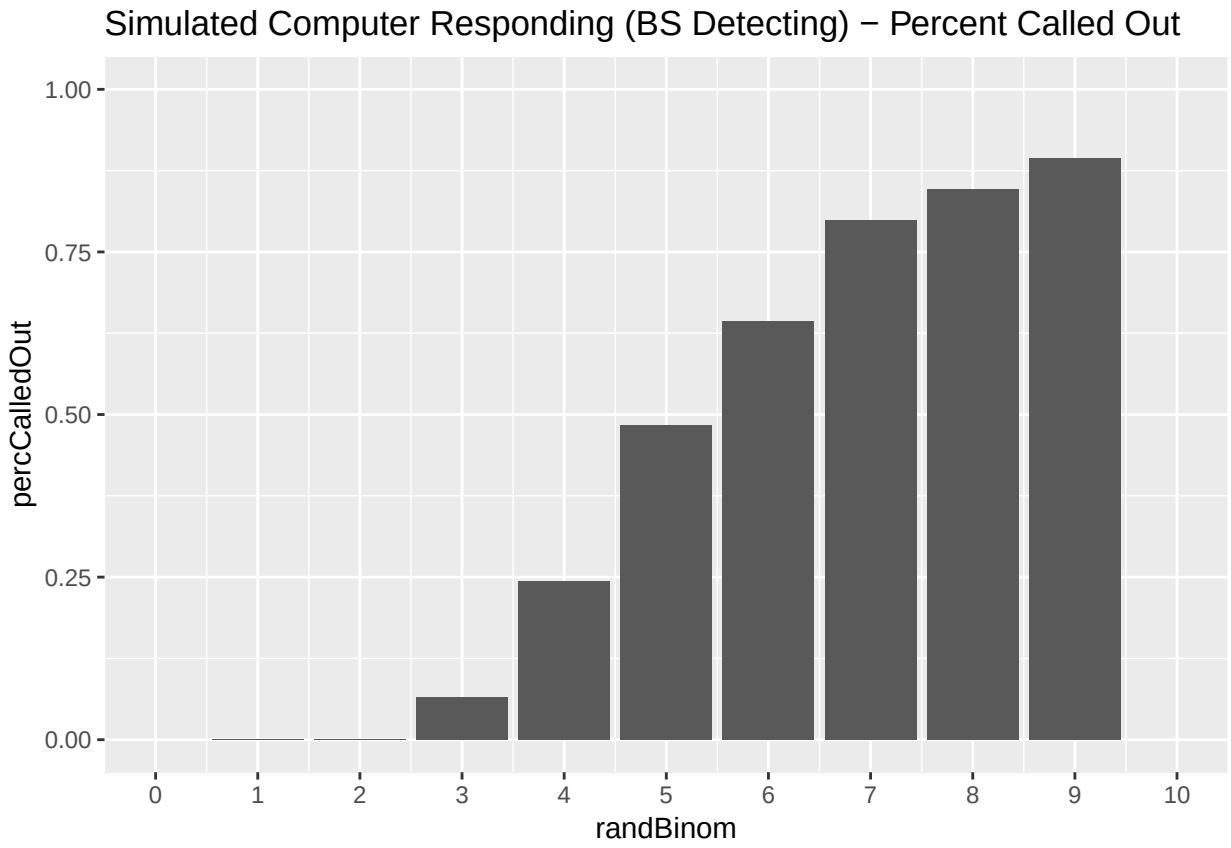
```
## Warning: Removed 3 rows containing missing values (geom_point).
```



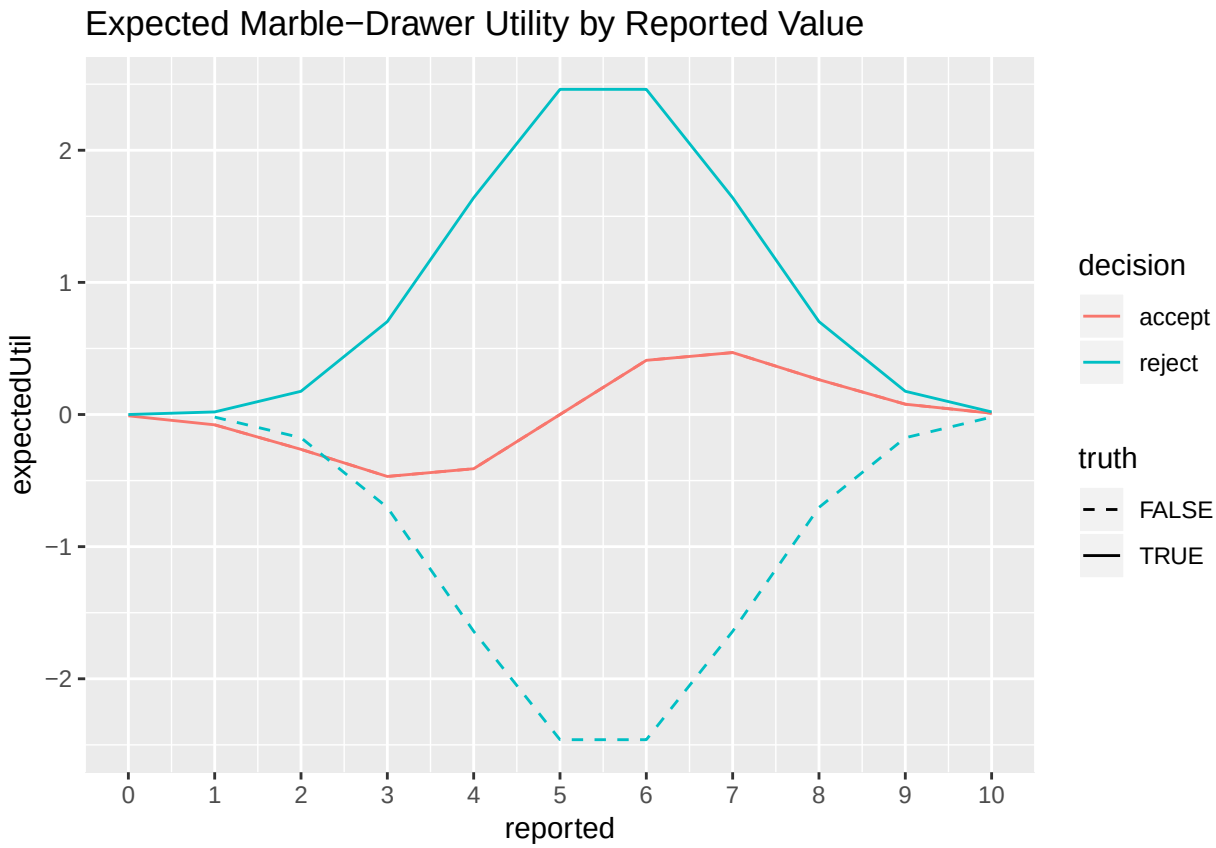

```
responderProbsBSDetect.sim <- data.frame(randBinom = rbinom(1000, 10, 0.5), rand = runif(1000))
responderProbsBSDetect.sim <- responderProbsBSDetect.sim %>%
  mutate(probCallOut = pbinom(randBinom, 10, 0.5) - (pbinom(5, 10, 0.5)-.5),
         calledOut = rand < probCallOut)

ggplot(responderProbsBSDetect.sim, aes(x=randBinom, y=probCallOut, colour=calledOut)) +
  geom_jitter() +
  scale_x_continuous(limits=c(0,10), breaks=0:10) +
  ggtitle("Simulated Computer Responding (BS Detecting)")
```




```
responderProbsBSDetect.sim %>%
  group_by(randBinom) %>%
  summarise(percCalledOut = sum(calledOut==TRUE)/n()) %>%
  ggplot(aes(x=randBinom, y=percCalledOut)) +
  geom_bar(stat="identity") +
  scale_x_continuous(limits=c(0,10), breaks=0:10) +
  scale_y_continuous(limits=c(0,1)) +
  ggtitle("Simulated Computer Responding (BS Detecting) - Percent Called Out")
```




```
responderProbsBD <- responderProbs %>%
  filter(reported >= real) %>%
  group_by(real) %>%
  mutate(truth = real == reported,
         sumProb = sum(binom),
         normalized = binom/sumProb,
         accept = utilDiffAccept * binom,
         reject = ifelse(truth, utilDiffReject.truth * binom, utilDiffReject.lie * binom))

responderProbsBD %>%
  group_by(reported, binom, cbinom, truth, accept) %>%
  summarise(reject = mean(reject)) %>%
  gather("decision", "expectedUtil",5:6) %>%
  ggplot(aes(x=reported, y=expectedUtil, colour=decision, linetype=truth)) +
  geom_line(stat="identity") +
  scale_x_continuous(breaks=pretty(reported, n=10)) +
  scale_linetype_manual(values=c(2,1)) +
  ggtitle("Expected Marble-Drawer Utility by Reported Value")
```




```
# responderProbsBD %>%
#   group_by(reported, binom, cbinom, accept) %>%
#   summarise(reject = mean(reject)) %>%
#   gather("decision", "expectedUtil",4:5) %>%
#   ggplot(aes(x=reported, y=expectedUtil, colour=decision)) +
#   geom_line(stat="identity") +
#   scale_x_continuous(breaks=pretty(reported, n=10)) +
#   ggtitle("Expected Marble-Drawer Utility by Reported Value, Averaging Truths + Lies")
```