```
# Part D

library(rootSolve)

# d.i.

# calculate ratios based on a feed on 100:300 PCO2:PH2

conversion <- conversion %>%
  mutate("Diox" = (P/4) * (1 - (conversion / 100))) %>%
  mutate("Hyd" = (P/4) * (3 - 3 * (conversion / 100))) %>%
  mutate("MeOH" = (P/4) * (conversion / 100)) %>%
  mutate("Wat" = (P/4) * (conversion / 100)) %>%
  mutate("Total" = Diox + Hyd + MeOH + Wat) %>%
  mutate("Kp" = (MeOH * Wat) / (Diox * Hyd^3)) %>%
  mutate("log10Kp" = log10(Kp)) %>%
  mutate("lnKp" = log(Kp))

KpPlot <- ggplot(data = conversion) +
  geom_line(mapping = aes(x = temp, y = log10Kp, color = pressure)) +
  labs(title = "Figure 8",
       subtitle = "Equilibrium Constant (Pressure) for Reaction One from
Conversion Data") +
  xlab("Temperature (Celcius)") +
  ylab("Kp (atm^-2) Log10 Scale")

ggsave("pics/Figure8.pdf")

# d.ii.

gasLaw$a <- gasLaw$a * 0.986 # bar to atm

R2 <- 0.08206 # L*atm / K*mol

vanDerWaals <- function(n, temp, P, a, b) {
  zero <-  ((P + a * (n * n)) * ((1 / n) - b)) - R2 * temp
  return(zero)
}

press1 <- conversion %>%
  filter(pressure == "1 atm") %>%
  select(Diox, Hyd, MeOH, Wat)
press10 <- conversion %>%
  filter(pressure == "10 atm") %>%
  select(Diox, Hyd, MeOH, Wat)
press50 <- conversion %>%
  filter(pressure == "50 atm") %>%
  select(Diox, Hyd, MeOH, Wat)
press100 <- conversion %>%
  filter(pressure == "100 atm") %>%
  select(Diox, Hyd, MeOH, Wat)

temp <- seq(100, 400, 25)
zed <- numeric(length(temp))
```

```
nComp1 <- data.frame(temp, "nDiox" = zed, "nHyd" = zed, "nMeOH" = zed, "nWat"
= zed) %>%
  mutate("pressure" = rep("1 atm", length(temp))) %>%
  mutate("P" = rep(1,length(temp)))
nComp10 <- nComp1 %>%
  mutate("pressure" = rep("10 atm", length(temp))) %>%
  mutate("P" = rep(10,length(temp)))
nComp50 <- nComp1 %>%
  mutate("pressure" = rep("50 atm", length(temp))) %>%
  mutate("P" = rep(50,length(temp)))
nComp100 <- nComp1 %>%
  mutate("pressure" = rep("100 atm", length(temp))) %>%
  mutate("P" = rep(100,length(temp)))

gasRange <- c(0, 1000)

for (i in 1:4) {
  for (j in 1:4) {
    # j = 1 for CO2, 2 for H2, 3 for CH3OH, 4 for H2O
    for (k in 1:length(temp)) {
      if (i == 1) {
        gassy <- function(n) {
          vanDerWaals(n, (temp[k] + 273), press1[k,j], gasLaw[j,2], gasLaw[j,
3])
        }
        nComp1[k, j + 1 ] <- min(uniroot.all(gassy, interval = gasRange))
      } else if (i == 2) {
        gassy <- function(n) {
          vanDerWaals(n, (temp[k] + 273), press10[k,j], gasLaw[j,2], gasLaw[j,
3])
        }
        nComp10[k, j + 1] <- min(uniroot.all(gassy, interval = gasRange))
      } else if (i == 3) {
        gassy <- function(n) {
          vanDerWaals(n, (temp[k] + 273), press50[k,j], gasLaw[j,2], gasLaw[j,
3])
        }
        nComp50[k, j + 1] <- min(uniroot.all(gassy, interval = gasRange))
      } else {
        gassy <- function(n) {
          vanDerWaals(n, (temp[k] + 273), press100[k,j], gasLaw[j,2],
gasLaw[j,3])
        }
        nComp100[k, j + 1] <- min(uniroot.all(gassy, interval = gasRange))
      }
    }
  }
}

nComp1 <- nComp1 %>%
  mutate("K" = (nMeOH * nWat) / (nDiox * nHyd * nHyd * nHyd)) %>%
  mutate("log10K" = log10(K))

nComp10 <- nComp10 %>%
  mutate("K" = (nMeOH * nWat) / (nDiox * nHyd * nHyd * nHyd)) %>%
  mutate("log10K" = log10(K))

nComp50 <- nComp50 %>%
  mutate("K" = (nMeOH * nWat) / (nDiox * nHyd * nHyd * nHyd)) %>%
  mutate("log10K" = log10(K))

nComp100 <- nComp100 %>%
  mutate("K" = (nMeOH * nWat) / (nDiox * nHyd * nHyd * nHyd)) %>%
  mutate("log10K" = log10(K))

nComp <- rbind(nComp1, nComp10, nComp50, nComp100)

KPlot <- ggplot(data = nComp) +
  geom_line(mapping = aes(x = temp, y = log10K, color = pressure)) +
  labs(title = "Figure 9",
       subtitle = "Equilibrium Constant for Reaction One from Conversion
Data") +
  xlab("Temperature (Celcius)") +
  ylab("K (L mol^-2) Log10 Scale")

ggsave("pics/Figure9.pdf")

# d.iv.

nComp <- nComp %>%
  mutate("lnK" = log(K))

lnKPlot <- ggplot(data = nComp %>% filter(temp == 100)) +
  geom_line(mapping = aes(x = P, y = lnK)) +
  labs(title = "Figure 10",
       subtitle = "How Equilibrium Depends on Pressure") +
  xlab("Pressure (atm)") +
  ylab("K (L mol^-2) Ln Scale")

ggsave("pics/Figure10.pdf")

dlnKdP <- (29.43 - 22.53) / (100 - 10)
k1 <- 8.206e-5
dv = -dlnKdP * k1 * (273 + 100)
```