In the previous post (<https://statcompute.wordpress.com/2018/01/28/modeling-lgd-with-proportional-odds-model>), We saw how to use Proportional Odds Models in the LGD model development. In particular, I specifically mentioned that we would estimate a sub-model, which can be Gamma or Simplex regression, to project the conditional mean for LGD values in the (0, 1) range. However, it is worth pointing out that, if we would define a finer LGD segmentation, the necessity of this sub-model is completely optional. A standalone Proportional Odds Model without any sub-model is more than sufficient to serve the purpose of stress testing, e.g. CCAR.

In the example below, I will define 5 categories based upon LGD values in the [0, 1] range, estimate a Proportional Odds Model as usual, and then demonstrate how to apply the model outcome in the setting of stress testing with the stressed model input, e.g. LTV.

First of all, I defined 5 instead of 3 categories for LGD values, as shown below. Nonetheless, we could use a even finer category definition in practice to achieve a more accurate outcome.

df <- read.csv("lgd.csv")

df$lgd <- round(1 - df$Recovery\_rate, 4)

l1 <- c(-Inf, 0, 0.0999, 0.4999, 0.9999, Inf)

l2 <- c("A", "B", "C", "D", "E")

df$lgd\_cat <- cut(df$lgd, breaks = l1, labels = l2, ordered\_result = T)

summary(df$lgd\_cat)

m1 <- ordinal::clm(lgd\_cat ~ LTV, data = df)

#Coefficients:

# Estimate Std. Error z value Pr(>|z|)

#LTV 2.3841 0.1083 22.02 <2e-16 \*\*\*

#

#Threshold coefficients:

# Estimate Std. Error z value

#A|B 0.54082 0.07897 6.848

#B|C 2.12270 0.08894 23.866

#C|D 3.18098 0.10161 31.307

#D|E 4.80338 0.13174 36.460

After the model estimation, it is straightforward to calculate the probability of each LGD category. The only question remained is how to calculate the LGD projection for each individual account as well as for the whole portfolio. In order to calculate the LGD projection, we need two factors, namely the probability and the expected mean of each LGD category, such that  
 **Estimated\_LGD = SUM\_i [Prob(category i) \* LGD\_Mean(category i)], where i = A, B, C, D, and E in this particular case.**  
The calculation is shown below with the estimated LGD = 0.23 that is consistent with the actual LGD = 0.23 for the whole portfolio.

prob\_A <- exp(df$LTV \* (-m1$beta) + m1$Theta[1]) / (1 + exp(df$LTV \* (-m1$beta) + m1$Theta[1]))

prob\_B <- exp(df$LTV \* (-m1$beta) + m1$Theta[2]) / (1 + exp(df$LTV \* (-m1$beta) + m1$Theta[2])) - prob\_A

prob\_C <- exp(df$LTV \* (-m1$beta) + m1$Theta[3]) / (1 + exp(df$LTV \* (-m1$beta) + m1$Theta[3])) - prob\_A - prob\_B

prob\_D <- exp(df$LTV \* (-m1$beta) + m1$Theta[4]) / (1 + exp(df$LTV \* (-m1$beta) + m1$Theta[4])) - prob\_A - prob\_B - prob\_C

prob\_E <- 1 - exp(df$LTV \* (-m1$beta) + m1$Theta[4]) / (1 + exp(df$LTV \* (-m1$beta) + m1$Theta[4]))

pred <- data.frame(prob\_A, prob\_B, prob\_C, prob\_D, prob\_E)

sum(apply(pred, 2, mean) \* aggregate(df['lgd'], df['lgd\_cat'], mean)[2])

#[1] 0.2262811

One might be wondering how to apply the model outcome with simple averages in stress testing that the model input is stressed, e.g. more severe, and might be also concerned about the lack of model sensitivity. In the demonstration below, let’s stress the model input LTV by 50% and then evaluate the stressed LGD.

df$LTV\_ST <- df$LTV \* 1.5

prob\_A <- exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[1]) / (1 + exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[1]))

prob\_B <- exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[2]) / (1 + exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[2])) - prob\_A

prob\_C <- exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[3]) / (1 + exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[3])) - prob\_A - prob\_B

prob\_D <- exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[4]) / (1 + exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[4])) - prob\_A - prob\_B - prob\_C

prob\_E <- 1 - exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[4]) / (1 + exp(df$LTV\_ST \* (-m1$beta) + m1$Theta[4]))

pred\_ST <- data.frame(prob\_A, prob\_B, prob\_C, prob\_D, prob\_E)

sum(apply(pred\_ST, 2, mean) \* aggregate(df['lgd'], df['lgd\_cat'], mean)[2])

#[1] 0.3600153

As shown above, although we only use a simple averages as the expected mean for each LGD category, the overall LGD still increases by ~60%. The reason is that, with the more stressed model input, the Proportional Odds Model is able to push more accounts into categories with higher LGD. For instance, the output below shows that, if LTV is stressed by 50% overall, ~146% more accounts would roll into the most severe LGD category without any recovery.

apply(pred\_ST, 2, mean) / apply(pred, 2, mean)

# prob\_A prob\_B prob\_C prob\_D prob\_E

#0.6715374 0.7980619 1.0405573 1.4825803 2.4639293