Variational Model Diagnostics Significance Testing of Rq3 Groups

Jeffrey M. Young

16 May 2020

Hello, in this walk-through we'll perform the significance testing on the financial and automotive datasets in light of the non-normality results from the `rq1_rq3` walk through. We are assuming you have read that walk-through first.

Preliminaires

First, libraries, note that I have silenced the shadowing warnings from R:

```
library(ggplot2) #plotting
library(dplyr)   #dataframe manipulation
library(tidyr)   #dataframe manipulation
library(broom)   #for the tidy function
library(scales)  #for scientific function
```

Let's load the data from the `counts` and `diag` files. Most of this should be pretty familiar.

```
## files
finFile        <- "../../data/fin_diag.csv"
autoFile       <- "../../data/auto_diag.csv"
finCountsFile  <- "../../data/fin_counts.csv"
autoCountsFile <- "../../data/auto_counts.csv"

## read the csv and add data column
finCountData  <- read.csv(file=finCountsFile) %>% mutate(data = "Fin")
autoCountData <- read.csv(file=autoCountsFile) %>% mutate(data = "Auto")

## Munge the datasets
autoRatio <- autoCountData %>%
  ## perform the following verbs with respect to variants
  group_by(Variants) %>%
  ## find raw count of Satisfiable models by variant
  count(Satisfiable) %>%
  ## turn the long data into wide data
  pivot_wider(names_from=Satisfiable, values_from=n) %>%
  ## calclulate the unsat ratio
  mutate(UnSatRatio = UnSat / (UnSat + Sat)) %>%
  ## finally replace NAs from calculations with 0s
  replace_na(list(UnSatRatio = 0, UnSat = 0))

## repeat for the financial dataset
finRatio <- finCountData %>%
  group_by(Variants) %>%
  count(Satisfiable) %>%
  pivot_wider(names_from=Satisfiable, values_from=n) %>%
```

```
  mutate(UnSatRatio = signif(UnSat / (UnSat + Sat), 3)) %>%
  replace_na(list(UnSatRatio = 0, UnSat = 0))

## Read in the model diagnostics files and calculate the unchanged ratio
## then merge with the previous data frames
finDF <- read.csv(file=finFile) %>%
  mutate(data = "Fin", UnchangedRatio = NumUnchange /NumFeatures ) %>%
  merge(finRatio)

autoDF <- read.csv(file=autoFile) %>%
  mutate(data = "Auto", UnchangedRatio = NumUnchange / NumFeatures) %>%
  merge(autoRatio)
```

And we can view the datasets:

```
## for fin
str(finDF)
```

```
## 'data.frame':    10 obs. of  10 variables:
##  $ Variants      : int  2 4 8 16 32 64 128 256 512 1024
##  $ Config        : Factor w/ 10 levels "V1","V1*V2","V1*V2*V3",..: 1 2 3 4 5 6 7 8 9 10
##  $ NumFeatures   : int  1072 1075 1079 1079 1081 1081 1084 1084 1084 1084
##  $ NumUnchange   : int  1005 976 971 956 940 934 919 880 857 851
##  $ MaxClause     : int  1 2 4 8 16 32 64 65 66 132
##  $ data          : chr  "Fin" "Fin" "Fin" "Fin" ...
##  $ UnchangedRatio: num  0.938 0.908 0.9 0.886 0.87 ...
##  $ Sat           : int  2 3 5 9 17 33 65 66 67 133
##  $ UnSat         : num  0 1 3 7 15 31 63 190 445 891
##  $ UnSatRatio    : num  0 0.25 0.375 0.438 0.469 0.484 0.492 0.742 0.869 0.87
```

```
summary(finDF)
```

```
##     Variants                   Config   NumFeatures    NumUnchange
##  Min.   :   2.0   V1              :1   Min.   :1072   Min.   : 851.0
##  1st Qu.:  10.0   V1*V2           :1   1st Qu.:1079   1st Qu.: 889.8
##  Median :  48.0   V1*V2*V3        :1   Median :1081   Median : 937.0
##  Mean   : 204.6   V1*V2*V3*V4     :1   Mean   :1080   Mean   : 928.9
##  3rd Qu.: 224.0   V1*V2*V3*V4*V5  :1   3rd Qu.:1084   3rd Qu.: 967.2
##  Max.   :1024.0   V1*V2*V3*V4*V5*V6:1   Max.   :1084   Max.   :1005.0
##                   (Other)         :4
##    MaxClause          data           UnchangedRatio        Sat
##  Min.   :  1.00   Length:10          Min.   :0.7851   Min.   :  2.00
##  1st Qu.:  5.00   Class :character   1st Qu.:0.8208   1st Qu.:  6.00
##  Median : 24.00   Mode  :character   Median :0.8668   Median : 25.00
##  Mean   : 39.00                      Mean   :0.8600   Mean   : 40.00
##  3rd Qu.: 64.75                      3rd Qu.:0.8964   3rd Qu.: 65.75
##  Max.   :132.00                      Max.   :0.9375   Max.   :133.00
##
##      UnSat         UnSatRatio
##  Min.   :  0.0   Min.   :0.0000
##  1st Qu.:  4.0   1st Qu.:0.3907
##  Median : 23.0   Median :0.4765
##  Mean   :164.6   Mean   :0.4989
##  3rd Qu.:158.2   3rd Qu.:0.6795
##  Max.   :891.0   Max.   :0.8700
```

```
##
```

and

```
## for auto
str(autoDF)
```

```
## 'data.frame':    4 obs. of  10 variables:
##  $ Variants      : int  2 4 8 16
##  $ Config        : Factor w/ 4 levels "V1","V1*V2","V1*V2*V3",..: 1 2 3 4
##  $ NumFeatures   : int  23300 23858 24054 24054
##  $ NumUnchange   : int  21159 22755 22746 22592
##  $ MaxClause     : int  1 2 3 7
##  $ data          : chr  "Auto" "Auto" "Auto" "Auto"
##  $ UnchangedRatio: num  0.908 0.954 0.946 0.939
##  $ Sat           : int  2 3 4 8
##  $ UnSat         : num  0 1 4 8
##  $ UnSatRatio    : num  0 0.25 0.5 0.5
```

```
summary(autoDF)
```

```
##     Variants            Config   NumFeatures     NumUnchange      MaxClause
##  Min.   : 2.0   V1         :1   Min.   :23300   Min.   :21159   Min.   :1.00
##  1st Qu.: 3.5   V1*V2      :1   1st Qu.:23718   1st Qu.:22234   1st Qu.:1.75
##  Median : 6.0   V1*V2*V3   :1   Median :23956   Median :22669   Median :2.50
##  Mean   : 7.5   V1*V2*V3*V4:1   Mean   :23816   Mean   :22313   Mean   :3.25
##  3rd Qu.:10.0                   3rd Qu.:24054   3rd Qu.:22748   3rd Qu.:4.00
##  Max.   :16.0                   Max.   :24054   Max.   :22755   Max.   :7.00
##      data           UnchangedRatio        Sat           UnSat
##  Length:4           Min.   :0.9081   Min.   :2.00   Min.   :0.00
##  Class :character   1st Qu.:0.9314   1st Qu.:2.75   1st Qu.:0.75
##  Mode  :character   Median :0.9424   Median :3.50   Median :2.50
##                     Mean   :0.9367   Mean   :4.25   Mean   :3.25
##                     3rd Qu.:0.9477   3rd Qu.:5.00   3rd Qu.:5.00
##                     Max.   :0.9538   Max.   :8.00   Max.   :8.00
##    UnSatRatio
##  Min.   :0.0000
##  1st Qu.:0.1875
##  Median :0.3750
##  Mean   :0.3125
##  3rd Qu.:0.5000
##  Max.   :0.5000
```

To plot we'll merge this into a single data frame and make a line + scatter plot:

```
df <- rbind(finDF, autoDF)

## custom breaks by dataset again
breaksRq1 <- function(x) {
  if (max(x) > 16) {
    2^(1:10)
  } else {
    2^(1:4)}
}

## define the x-axis for both scatter and line plots to be variants
ggplot(df, aes(x=Variants)) +
```

```
## we add `geom` layers to the plots with different y-axis variables
## we do not require a second y-axis because we've normalized the y-axis to ratios
geom_point(aes(y=UnchangedRatio,color="% Features Unchanged"),size=2) +
geom_point(aes(y=UnSatRatio,color="% Unsatisfiable Models"),size=2) +

## for the line plot we change the line type based on what is being plotted
## and increase the size
geom_line(aes(y=UnchangedRatio, color="% Features Unchanged"),
          linetype="dashed", size=1.1) +
geom_line(aes(y=UnSatRatio, color="% Unsatisfiable Models"),
          linetype="dotdash", size=1.1) +

## make x-axis log scale, use our custom breaks function and facet by dataset
scale_x_log10(breaks=breaksRq1) +
facet_wrap(. ~ data, scales="free_x") +

## niceties such as legend position, and theming
theme_classic() +
scale_y_continuous(breaks=seq(0,1,0.1)) +
theme(legend.position=c(0.80,0.15),
      legend.text=element_text(size=10),
      legend.key.size = unit(.55,'cm')) +
guides(color=guide_legend("")) +
ylab("Percent of Total") +
ggtitle("Ratio of unsatisfiable variants, and constant Features in V-Model")
```