Positivity Assumption Evaluation

Josiah Davis

4/12/2017

Approach

The goal of this document is to select the number of splits to use for the continuous variables contained in W with the objective of removing a minimal amount of observations due to violations of the positivity assumption. There are eight covariates W in the data which are discussed.

1. satisfaction_level
2. last_evaluation
3. number_project
4. average_montly_hours
5. time_spend_company
6. Work_accident
7. promotion_last_5years
8. sales

There are three continuous numeric variables. Quantiles are used to split the variables into buckets of size $k = 2, 3, ..., 10$.
1. satisfaction_level
2. last_evaluation
4. average_montly_hours

There are two discrete numeric variables. Quantiles are used to split them into buckets of $k = 2, 3$.
3. number_project
5. time_spend_company

There are three variables that are not transformed. This because they are already binary or becuase they have no natural ordering:
6. Work_accident
7. promotion_last_5years
8. sales

Define functions

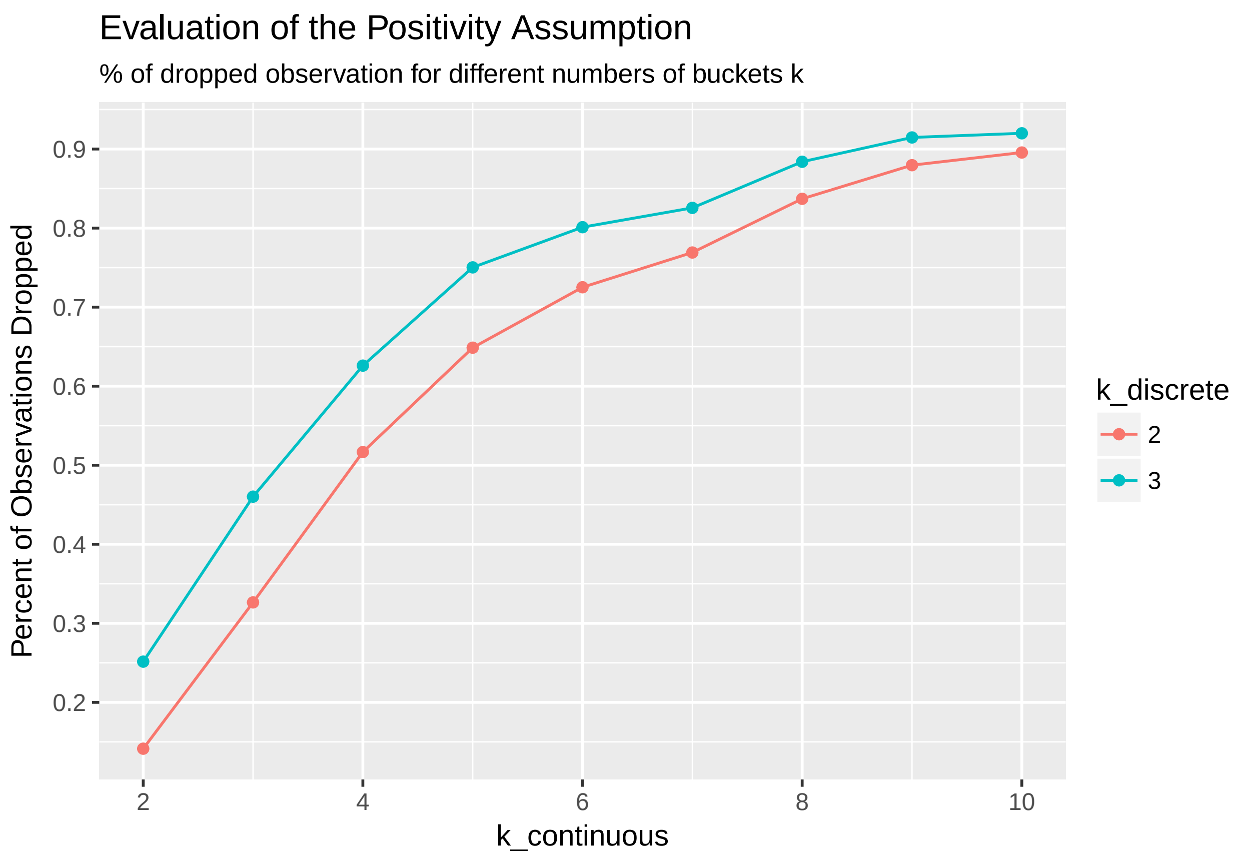
```
get_data <- function(dir){
  df <- read.csv(paste0(dir, 'HR_comma_sep_2.csv'))
  df <- df %>% filter(salary != 'medium')
  return(df)
}

split_var <- function(x, k){
  x_split <- cut(x, breaks = quantile(x, probs = seq(0, 1, 1/k)), include.lowest = TRUE)
  if(any(is.na(x_split))) stop('There are NA values in')
  return(factor(x_split, labels = 1:k))
}

count_drop_outs <- function(df, k_disc = 2, k_cont = 2){
  # Count the total number of observations removed
```

```r
  # when we remove strata with positivity violations

  n_pos_viol <- df %>%
  mutate(
    satisfaction_level_s = split_var(satisfaction_level, k_cont)
    , last_evaluation_s = split_var(last_evaluation, k_cont)
    , number_project_s = split_var(number_project, k_disc)
    , average_montly_hours_s = split_var(average_montly_hours, k_cont)
    , time_spend_company_s = split_var(time_spend_company, k_disc)
  ) %>%
  group_by(
    satisfaction_level_s
    , last_evaluation_s
    , number_project_s
    , average_montly_hours_s
    , time_spend_company_s
    , Work_accident
    , promotion_last_5years
    , sales
  ) %>%
  summarize(
    sal_h = sum(salary == 'high'),
    sal_l = sum(salary == 'low'),
    class_size = n()
  ) %>% ungroup() %>%
  filter(sal_h == 0 | sal_l == 0) %>%
  summarize(
    tot = sum(sal_h) + sum(sal_l)
  )
  return(unlist(n_pos_viol)[1])
}

format_df <- function(m){
  df <- as.data.frame(m)
  colnames(df) <- c('k_discrete', 'k_continuous', 'num_dropped')
  df$k_discrete <- factor(df$k_discrete)
  df$pct_dropped <- df$num_dropped / nrow(d)
  return(df)
}

count_over_k <- function(df){
  m <- matrix(nrow = 2*9, ncol = 3)
  i <- 1
  for (k_disc in 2:3){
    for (k_cont in 2:10){
      m[i,1:2] <- c(k_disc, k_cont)
      m[i,3] <- count_drop_outs(df, k_disc, k_cont)
      i <- i + 1
    }
  }
  return(m)
}
```

```
visualize <- function(df){
  plt <- ggplot(results, aes(x = k_continuous, y = pct_dropped
                           , group = k_discrete
                           , color = k_discrete)) +
  geom_line() +
  scale_y_continuous(breaks = seq(0, 1, by = .1)) +
  labs(y = 'Percent of Observations Dropped'
       , title = 'Evaluation of the Positivity Assumption'
       , subtitle = '% of dropped observation for different numbers of buckets k') +
  geom_point()
  return(plt)
}
```

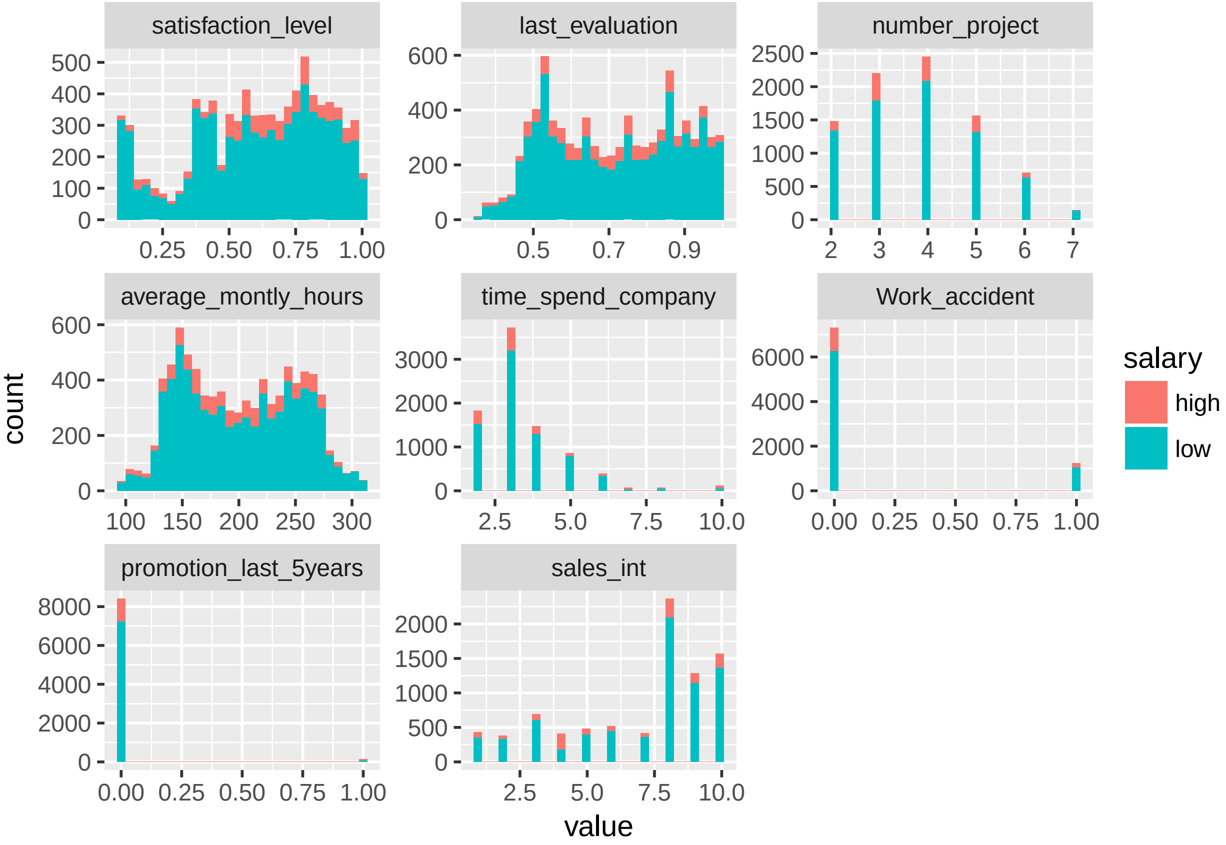
Run evaluation

```
library(dplyr)
library(ggplot2)

DIR <- '/Users/josiahdavis/Documents/Berkeley/PH252D/data/' # <= UPDATE AS NEEDED

# Read in data
d <- get_data(DIR)

# Count the results over a range of k values
raw_results <- count_over_k(d)

# Format results to make it easier to visualize
results <- format_df(raw_results)

# Create the data visusalizeion
visualize(results)
```



Other

```r
library(reshape2)
# Generate an exploratory graphic to illustrate how
# variables are distributed
d$subject <- 1:nrow(d)
d$sales_int <- as.integer(d$sales)
de <- melt(d, id.vars = c('subject', 'salary')
           , measure.vars = c('satisfaction_level', 'last_evaluation', 'number_project'
                              , 'average_montly_hours', 'time_spend_company', 'Work_accident'
                              , 'promotion_last_5years', 'sales_int'))
ggplot(de, aes(x = value, fill = salary)) +
  geom_histogram() +
  facet_wrap(~variable, scales = 'free')
```


satisfaction_level
last_evaluation
number_project
average_montly_hours
time_spend_company
Work_accident
promotion_last_5years
sales_int
count
value
salary
high
low
0
100
200
300
400
500
0.25
0.50
0.75
1.00
600
0.5
0.7
0.9
500
1000
1500
2000
2500
2
3
4
5
6
7
150
250
300
3000
2.5
5.0
7.5
10.0
4000
6000
0.00
8000