Predicting Revenue from Google Advertising Data

MATH2319 - Machine Learning
Course Project

Ben Cole - s3412349

Print Date: 26/05/2019

Contents

1 Phase 1 - Introduction, Cleaning, and Exploration

1.1 Outline

The prescribed data set contained advertising metrics provided by a prominent search engine. The data contained severald descriptive features pertaining to a range of information. Finally, the target feature was a measure of revenue associated with each of the observations.

1.1.1 Nature of the Data

The below is an exerpt from accompanying documentation about the dataset.

Features in this data set are as follows:

- companyId: Company ID of record (categorical)
- countryId: Country ID of record (categorical)
- deviceType: Device type of record (categorical corresponding to desktop, mobile, tablet)
- day: Day of record (integer between 1 (oldest) and 30 for train, 31 and 35 (most recent) for test)
- dow: Day of week of the record (categorical)
- price1, price2, price3: Price combination for the record set by the company (numeric)
- ad_area: area of advertisement (numeric)
- ad_ratio: ratio of advertisement's length to its width (numeric)
- requests, impression, cpc, ctr, viewability: Various metrics related to the record (numeric)
- ratio1, …, ratio5: Ratio characteristics related to the record (numeric)
- y (target feature): revenue-related metric (numeric)

1.1.1.1 Target Feature

The column/variable **y** was selected as the target feature in the dataset.

1.1.1.2 Descriptive Features

All other columns/variables in the dataset, as outlined above, were chosen as descriptive features.

1.2 Data Processing

1.2.1 Libraries

```r
library(pacman)                         ## for loading multiple packages

suppressMessages(p_load(character.only = T,
                        install = F,
                        c("tidyverse",  ## thanks Hadley
                          "lubridate",  ## for handling dates
                          "forcats",    ## for categorial variables, not for felines
                          "zoo",        ## some data cleaning capabilities
                          "lemon",      ## add ons for ggplot
                          "rvest",      ## scraping web pages
                          "knitr",      ## knitting to RMarkdown
                          "kableExtra", ## add ons for knitr tables
                          "scales",     ## quick and easy formatting prettynums
                          "grid",       ## for stacking ggplots
                          "gridExtra",  ## also for stacking ggplots
                          "e1071",      ## for skew and kurtosis
                          "janitor",    ## cleaning colnames
                          "beepr")))    ## plays a beep tone
```

Table 1: Sample of Advertising Data Frame

case_id	companyId	countryId	deviceType	day	dow	price1	price2	price3	ad_area	ad_ratio
181647	159	110	2	26	Wednesday	0.01	0.07	0.3277	0.0001	1.00000
15343	43	108	2	3	Monday	0.00	0.00	0.0000	7.5000	0.83333
201831	159	12	3	29	Saturday	0.17	0.66	1.3039	0.0001	1.00000
21450	126	77	3	4	Tuesday	0.65	10.42	10.4227	7.5000	0.83333
102559	43	56	1	15	Saturday	0.00	0.00	0.0000	9.0000	1.00000
18493	43	191	2	4	Tuesday	0.00	0.00	0.0000	0.0001	1.00000
180629	43	234	3	25	Tuesday	0.08	0.23	0.4652	6.5520	0.12363
125423	43	68	2	18	Tuesday	1.72	2.70	5.3996	8.7300	0.09278
29565	43	234	2	5	Wednesday	0.00	0.00	0.0000	2.8080	0.12821
125263	43	166	3	18	Tuesday	0.00	0.00	0.0000	0.0001	1.00000
62406	43	70	2	10	Monday	0.00	0.00	0.0000	0.0001	1.00000
62614	43	167	2	10	Monday	0.26	0.54	1.0884	7.5000	0.83333
37242	43	218	1	6	Thursday	0.00	0.00	0.0000	0.0001	1.00000
188347	43	189	1	27	Thursday	0.00	0.00	0.0000	0.0001	1.00000
100012	43	70	1	15	Saturday	0.00	0.00	0.0000	0.0001	1.00000
106663	43	3	2	16	Sunday	0.00	0.00	0.0000	0.0001	1.00000
14366	95	38	1	3	Monday	0.05	0.16	0.3100	7.5000	0.83333
120239	43	38	5	18	Tuesday	1.33	3.66	7.3215	0.0001	1.00000
209015	159	43	1	30	Sunday	0.00	0.00	0.0000	0.0001	1.00000
107539	43	70	2	16	Sunday	0.24	0.71	1.4243	18.0000	2.00000

1.2.2 Loading Data

```
advertising_train <- read_csv("advertising_train.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   dow = col_character()
## )

## See spec(...) for full column specifications.
```

```
sample_adv <- sample_n(advertising_train, 20)

kable_styling(kable(sample_adv[ , 1:(ncol(sample_adv)/2)],
                    caption = "Sample of Advertising Data Frame"),
              font_size = 8.5, latex_options = c("striped"),
              full_width = F)
```

```
kable_styling(kable(sample_adv[ , c(1, ((ncol(sample_adv)/2)+1):ncol(sample_adv))],
                    caption = "Sample of Advertising Data Frame (cont)"),
              font_size = 8.5, latex_options = c("striped"),
              full_width = F)
```

1.2.3 Classifying Data

Per the above feature definitions, the data was classified.

```
advertising_train$companyId <- as.factor(advertising_train$companyId)

advertising_train$countryId <- as.factor(advertising_train$countryId)

advertising_train$deviceType <- as.factor(advertising_train$deviceType)
```

Table 2: Sample of Advertising Data Frame (cont)

case_id	requests	impression	cpc	ctr	viewability	ratio1	ratio2	ratio3	ratio4	ratio5	y
181647	518	341	0.0388	0.0088	0.5124	0.9853	0.6569	1.0000	0.0000	0.0000	0.2840404
15343	125	57	0.0099	0.0175	0.8400	1.0000	0.9825	1.0000	0.0000	0.0000	0.1155405
201831	0	0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.5640816
21450	294	188	0.2198	0.0106	0.5057	0.8617	0.6223	0.4787	0.0000	0.5160	2.0681818
102559	301	301	0.1515	0.0033	0.3794	1.0000	0.8339	0.0831	0.2558	0.6611	0.5788462
18493	26	25	0.0004	0.2400	0.8667	1.0000	0.6800	1.0000	0.0000	0.0000	0.1081081
180629	2164	1285	1.0262	0.0008	0.6590	0.9704	0.9066	0.0000	0.8638	0.1362	0.4098881
125423	3352	1121	0.2645	0.0089	0.4237	0.6137	0.9384	1.0000	0.0000	0.0000	0.7592422
29565	0	0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2157692
125263	442	182	0.0185	0.0385	0.8085	0.9780	0.8791	0.0879	0.5385	0.3736	0.4721461
62406	28	25	0.0004	0.0800	1.0000	1.0000	0.9200	1.0000	0.0000	0.0000	0.0222222
62614	43306	14211	1.1679	0.0005	0.4963	0.6729	0.8702	1.0000	0.0000	0.0000	0.1619755
37242	0	0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500000
188347	93	93	0.0226	0.0323	1.0000	1.0000	0.8495	0.0323	0.2796	0.6882	0.6512658
100012	0	0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	4.4000000
106663	0	0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0250000
14366	0	0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0984871
120239	137	135	0.4547	0.0148	0.9565	0.7630	0.4741	1.0000	0.0000	0.0000	7.4450820
209015	86	59	0.0865	0.0169	0.7500	1.0000	0.4746	0.0000	0.7458	0.2542	1.3070968
107539	14631	1244	0.7363	0.0008	0.3724	0.4035	0.7460	1.0000	0.0000	0.0000	0.0693764

```r
advertising_train$dow <- as.factor(advertising_train$dow)

sapply(advertising_train, class)
```

```
##     case_id   companyId   countryId  deviceType         day         dow 
##   "numeric"    "factor"    "factor"    "factor"   "numeric"    "factor" 
##      price1      price2      price3     ad_area    ad_ratio    requests 
##   "numeric"   "numeric"   "numeric"   "numeric"   "numeric"   "numeric" 
##  impression         cpc         ctr viewability      ratio1      ratio2 
##   "numeric"   "numeric"   "numeric"   "numeric"   "numeric"   "numeric" 
##      ratio3      ratio4      ratio5           y 
##   "numeric"   "numeric"   "numeric"   "numeric"
```

1.2.4 Descriptive Statistics

1.2.4.1 Numeric Variables

```r
advertising_train_long_num <- select(advertising_train,
                                 colnames(advertising_train),
                                 -case_id, -countryId,
                                 -companyId, -deviceType,
                                 -dow)

advertising_train_long_num <- gather(advertising_train_long_num,
                                 key = "Variable",
                                 value = "Value")

summary_adv_num <- summarise(group_by(advertising_train_long_num,
                                  Variable),
                         "Mean" = mean(Value, na.rm = T),
                         "Std Dev" = sd(Value, na.rm = T),
                         "Min" = min(Value, na.rm = T),
                         "Q1" = quantile(Value, 0.25, na.rm = T),
```

Table 3: Summary Statistics of Numeric Variables

Variable	Mean	Std Dev	Min	Q1	Median	Q3	Max	Number of NA
ad_area	4.724	6.273	0.000	0.000	0.000	7.500	36.000	0.000
ad_ratio	0.923	0.482	0.083	0.833	1.000	1.000	5.000	0.000
cpc	0.178	0.707	0.000	0.000	0.016	0.125	132.534	0.000
ctr	0.033	0.093	0.000	0.000	0.002	0.012	2.000	0.000
day	15.791	8.386	1.000	9.000	16.000	23.000	30.000	0.000
impression	5,585.714	98,713.340	0.000	0.000	99.000	1,058.000	6,100,324.000	0.000
price1	0.438	1.281	0.000	0.000	0.010	0.190	14.690	0.000
price2	0.630	1.482	0.000	0.000	0.090	0.570	63.120	0.000
price3	0.932	1.840	0.000	0.000	0.295	0.986	78.900	0.000
ratio1	0.558	0.447	0.000	0.000	0.750	1.000	1.000	0.000
ratio2	0.491	0.414	0.000	0.000	0.627	0.896	1.027	0.000
ratio3	0.312	0.444	0.000	0.000	0.028	1.000	1.500	0.000
ratio4	0.131	0.240	0.000	0.000	0.000	0.164	1.077	0.000
ratio5	0.188	0.297	0.000	0.000	0.000	0.385	1.200	0.000
requests	8,678.997	122,347.229	0.000	0.000	147.000	1,633.000	6,701,924.000	0.000
viewability	0.378	0.366	0.000	0.000	0.332	0.716	7.000	0.000
y	0.847	1.391	0.000	0.150	0.419	0.959	47.060	0.000

```
                         "Median" = median(Value, na.rm = T),
                         "Q3" = quantile(Value, 0.75, na.rm = T),
                         "Max" = max(Value, na.rm = T),
                         "Number of NA" = sum(is.na(Value)))

kable_styling(kable(summary_adv_num,
                    digits = 3, format.args = list(nsmall = 3,
                                                   scientific = F,
                                                   big.mark = ","),
                    caption = "Summary Statistics of Numeric Variables"),
              font_size = 8.5, latex_options = c("striped"),
              full_width = F)
```

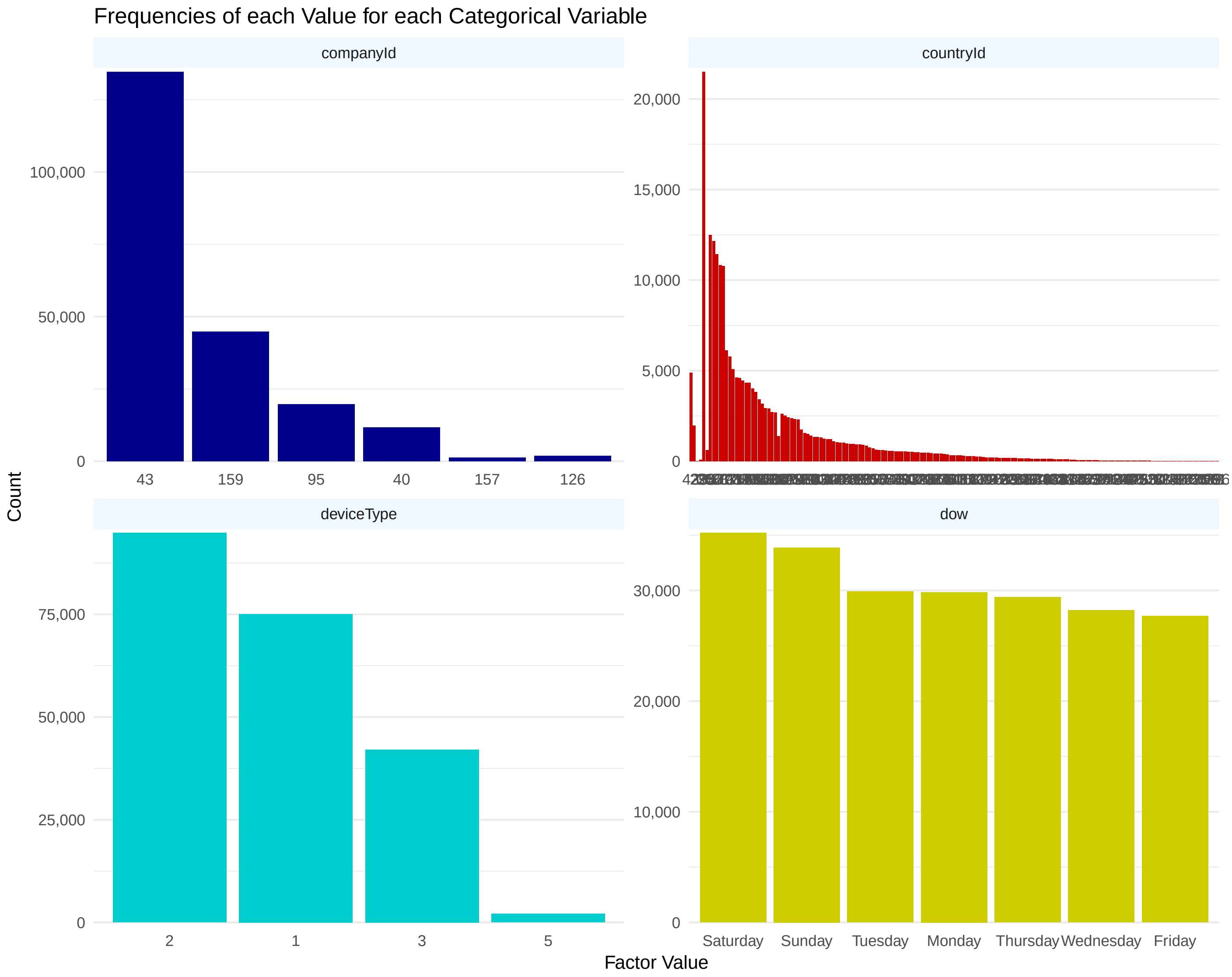
1.2.4.2 Categorical and Non-Numeric Variables

```
advertising_train_long_cat <- select(advertising_train,
                                     countryId,
                                     companyId, deviceType,
                                     dow)

advertising_train_long_cat <- gather(advertising_train_long_cat,
                                     key = "Variable",
                                     value = "Value")
```

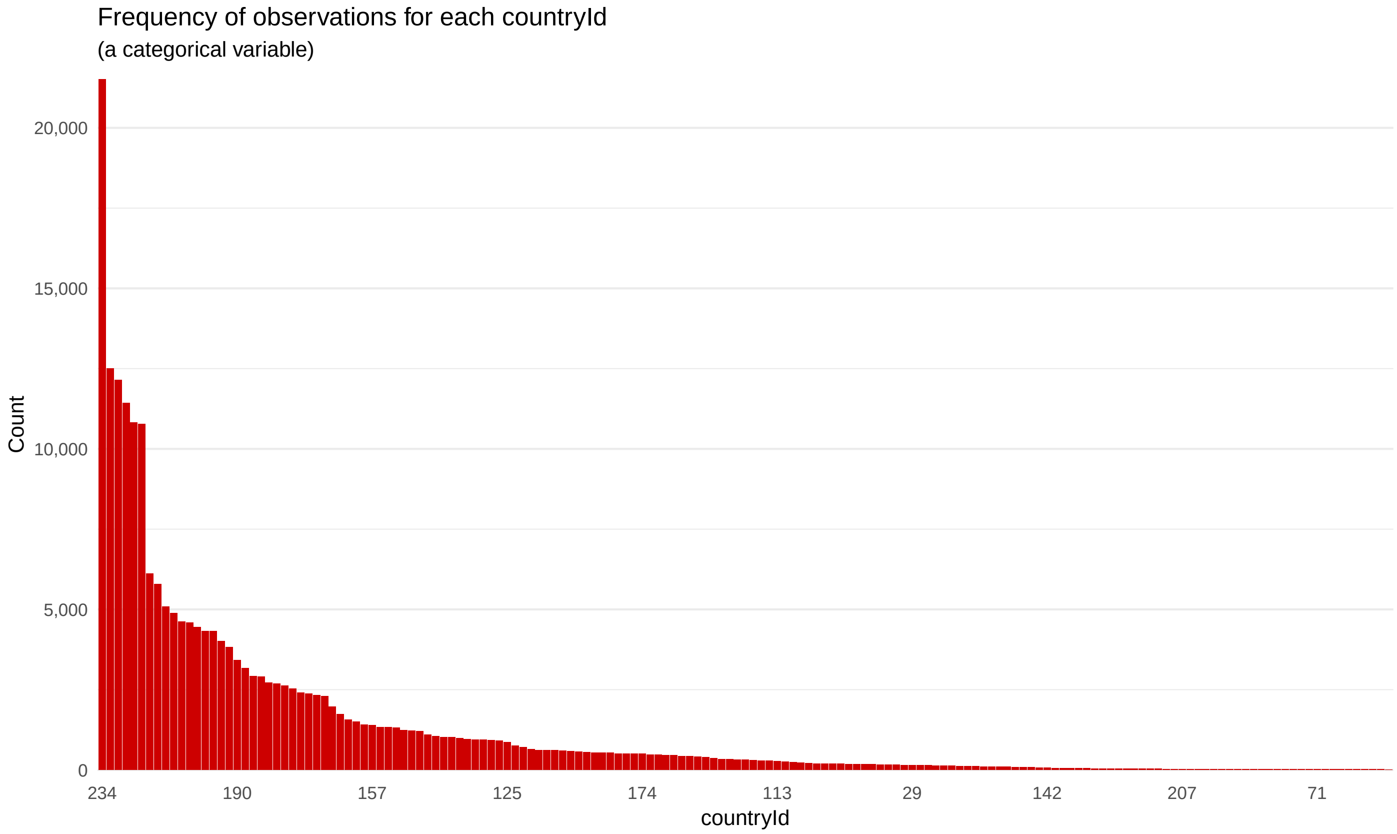
```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

```
advertising_train_long_cat$Variable <- as.factor(advertising_train_long_cat$Variable)

advertising_train_long_cat$Value <- as.factor(advertising_train_long_cat$Value)

ggplot(advertising_train_long_cat) +
  geom_bar(aes(x = fct_infreq(Value),
               fill = Variable),
           show.legend = F) +
```

```r
  facet_rep_wrap(~Variable,
                 repeat.tick.labels = T,
                 scales = "free") +
  scale_y_continuous(labels = comma,
                     expand = c(0.01, 0),
                     "Count") +
  scale_x_discrete("Factor Value") +
  scale_fill_manual(values = c("blue4", "red3", "cyan3", "yellow3")) +
  labs(title = "Frequencies of each Value for each Categorical Variable") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        strip.background = element_rect(fill = "aliceblue",
                                        colour = NA))
```



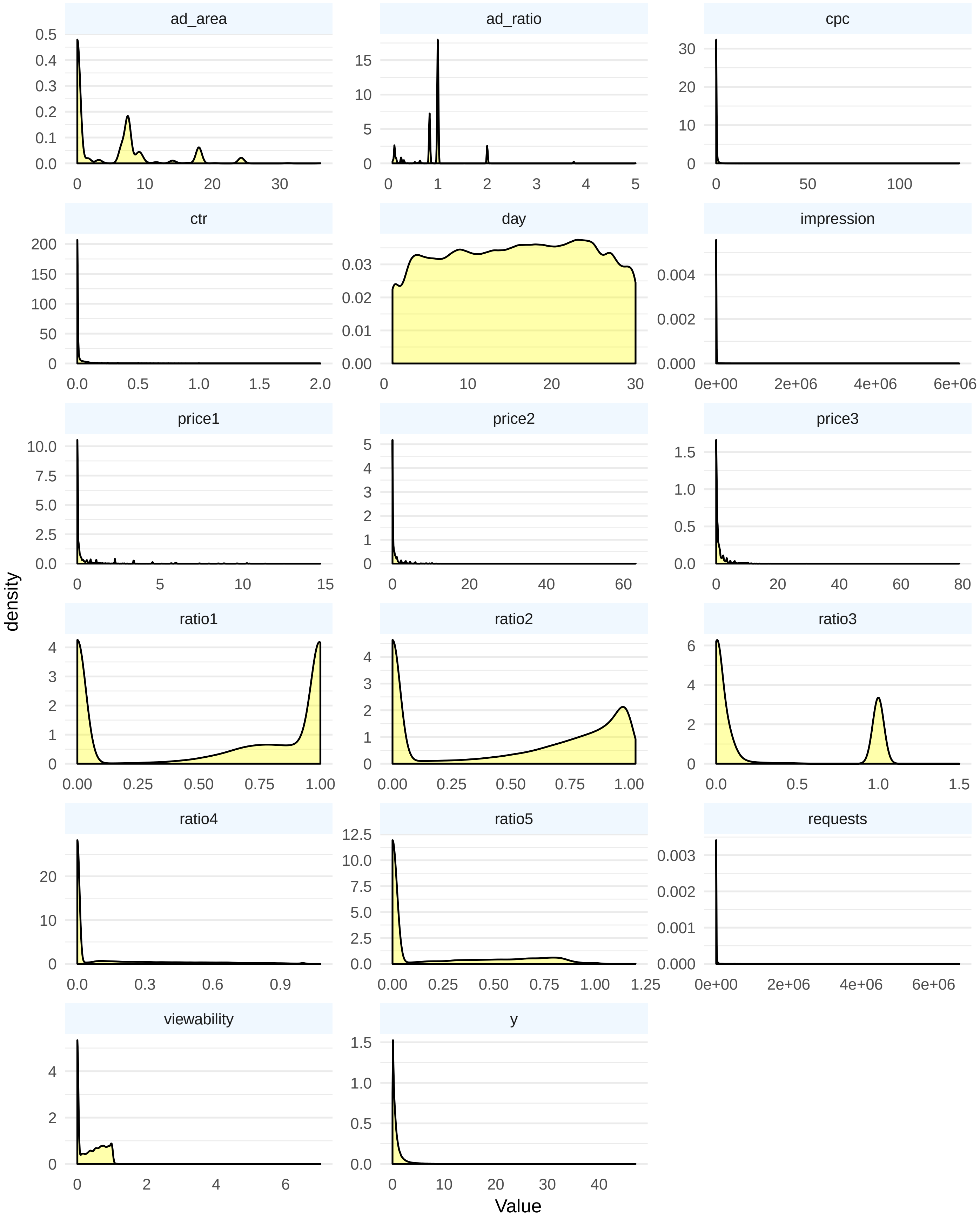
```r
country_labels <- levels(fct_infreq(advertising_train$countryId))[c(seq(1,
                                                  length(levels(fct_infreq(advertising_train$country
                                                  ceiling(length(levels(fct_infreq(advertising_trai

ggplot(advertising_train) +
  geom_bar(aes(x = fct_infreq(countryId)),
               fill = "red3") +
  scale_y_continuous(labels = comma,
                     expand = c(0.01, 0),
                     "Count") +
  scale_x_discrete(breaks = country_labels,
                   "countryId") +
  labs(title = "Frequency of observations for each countryId",
       subtitle = "(a categorical variable)") +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank())
```



1.2.5 Density Plots

1.2.5.1 Numeric Variables

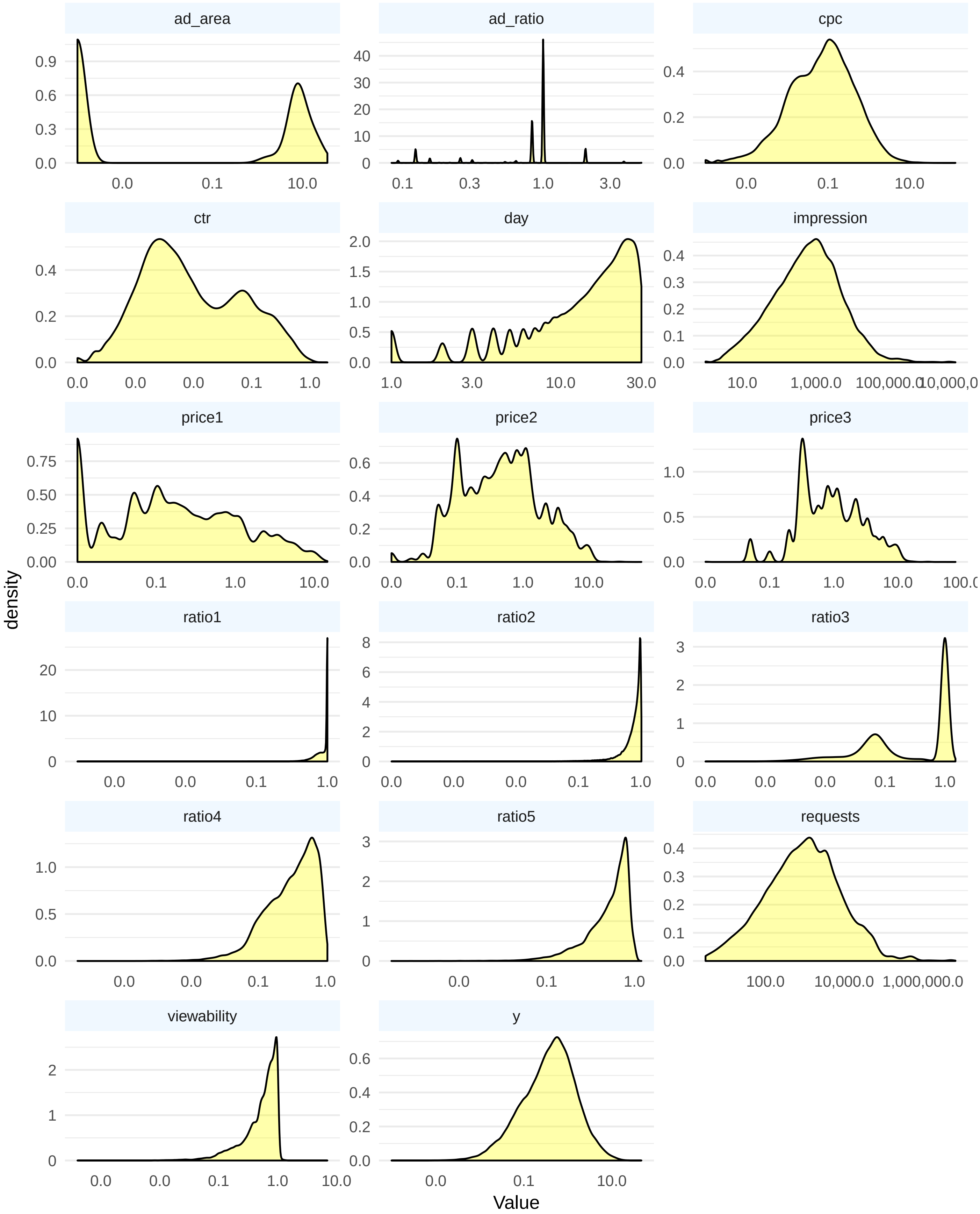
```
ggplot(advertising_train_long_num) +
  geom_density(aes(x = Value),
               fill = "yellow",
               alpha = 1/3) +
  facet_rep_wrap(~Variable,
                 repeat.tick.labels = T,
                 scales = "free",
                 ncol = 3) +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        strip.background = element_rect(fill = "aliceblue",
                                        colour = NA))
```



```r
ggplot(advertising_train_long_num) +
  geom_density(aes(x = Value),
               fill = "yellow",
               alpha = 1/3) +
  facet_rep_wrap(~Variable,
                 repeat.tick.labels = T,
                 scales = "free",
                 ncol = 3) +
  scale_x_log10(labels = comma_format(accuracy = 0.1)) +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        strip.background = element_rect(fill = "aliceblue",
                                        colour = NA))
```

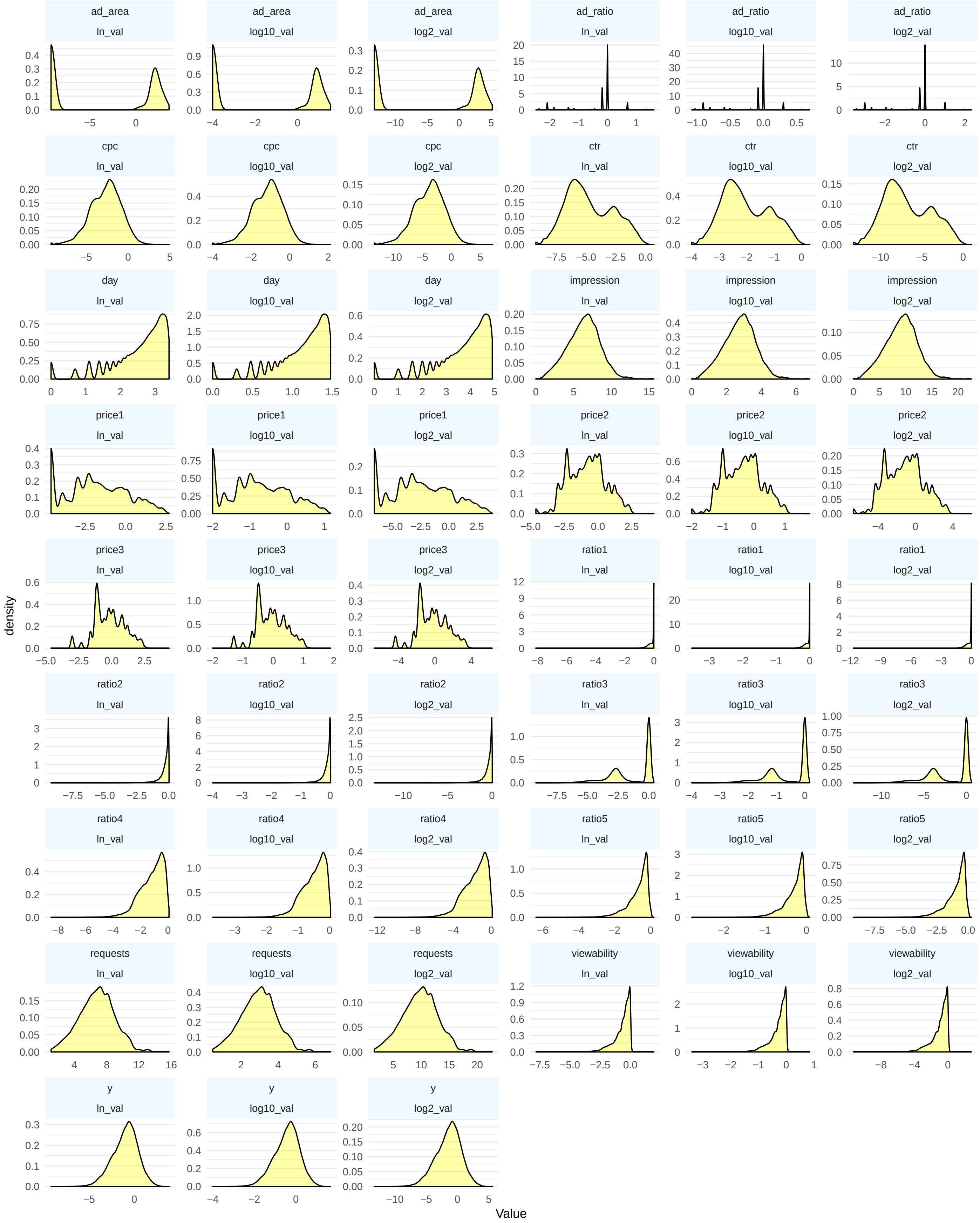
```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 1213004 rows containing non-finite values (stat_density).
```



```
advertising_train_long_num <- mutate(advertising_train_long_num,
                                 "log2_val" = log2(Value),
                                 "ln_val" = log(Value),
                                 "log10_val"= log10(Value))

log_advertising <- gather(select(advertising_train_long_num,
                                 Variable, log2_val, ln_val, log10_val),
                          log2_val, ln_val, log10_val,
                          key = "Transformation",
                          value = "Value")

ggplot(log_advertising) +
  geom_density(aes(x = Value),
               fill = "yellow",
               alpha = 1/3) +
  facet_rep_wrap(Variable ~ Transformation,
                 scales = "free",
                 ncol = 6) +
  theme_minimal() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        strip.background = element_rect(fill = "aliceblue",
                                        colour = NA))
```

```
## Warning: Removed 3639012 rows containing non-finite values (stat_density).
```


density
Value
ad_area
ln_val
0.4 0.3 0.2 0.1 0.0
−5 0
ad_area
log10_val
0.9 0.6 0.3 0.0
−4 −2 0
ad_area
log2_val
0.3 0.2 0.1 0.0
−10 −5 0 5
ad_ratio
ln_val
20 15 10 5 0
−2 −1 0 1
ad_ratio
log10_val
40 30 20 10 0
−1.0 −0.5 0.0 0.5
ad_ratio
log2_val
10 5 0
−2 0 2
cpc
ln_val
0.20 0.15 0.10 0.05 0.00
−5 0 5
cpc
log10_val
0.4 0.2 0.0
−4 −2 0 2
cpc
log2_val
0.15 0.10 0.05 0.00
−10 −5 0 5
ctr
ln_val
0.20 0.15 0.10 0.05 0.00
−7.5 −5.0 −2.5 0.0
ctr
log10_val
0.4 0.2 0.0
−4 −3 −2 −1 0
ctr
log2_val
0.15 0.10 0.05 0.00
−10 −5 0
day
ln_val
0.75 0.50 0.25 0.00
0 1 2 3
day
log10_val
2.0 1.5 1.0 0.5 0.0
0.0 0.5 1.0 1.5
day
log2_val
0.6 0.4 0.2 0.0
0 1 2 3 4 5
impression
ln_val
0.20 0.15 0.10 0.05 0.00
0 5 10 15
impression
log10_val
0.4 0.3 0.2 0.1 0.0
0 2 4 6
impression
log2_val
0.10 0.05 0.00
0 5 10 15 20
price1
ln_val
0.4 0.3 0.2 0.1 0.0
−2.5 0.0 2.5
price1
log10_val
0.75 0.50 0.25 0.00
−2 −1 0 1
price1
log2_val
0.2 0.1 0.0
−5.0 −2.5 0.0 2.5
price2
ln_val
0.3 0.2 0.1 0.0
−5.0 −2.5 0.0 2.5
price2
log10_val
0.6 0.4 0.2 0.0
−2 −1 0 1
price2
log2_val
0.20 0.15 0.10 0.05 0.00
−4 0 4
price3
ln_val
0.6 0.4 0.2 0.0
−5.0 −2.5 0.0 2.5
price3
log10_val
1.0 0.5 0.0
−2 −1 0 1 2
price3
log2_val
0.4 0.3 0.2 0.1 0.0
−4 0 4
ratio1
ln_val
12 9 6 3 0
−8 −6 −4 −2 0
ratio1
log10_val
20 10 0
−3 −2 −1 0
ratio1
log2_val
8 6 4 2 0
−12 −9 −6 −3 0
ratio2
ln_val
3 2 1 0
−7.5 −5.0 −2.5 0.0
ratio2
log10_val
8 6 4 2 0
−4 −3 −2 −1 0
ratio2
log2_val
2.5 2.0 1.5 1.0 0.5 0.0
−10 −5 0
ratio3
ln_val
1.0 0.5 0.0
−7.5 −5.0 −2.5 0.0
ratio3
log10_val
3 2 1 0
−4 −3 −2 −1 0
ratio3
log2_val
1.00 0.75 0.50 0.25 0.00
−10 −5 0
ratio4
ln_val
0.4 0.2 0.0
−8 −6 −4 −2 0
ratio4
log10_val
1.0 0.5 0.0
−3 −2 −1 0
ratio4
log2_val
0.4 0.3 0.2 0.1 0.0
−12 −8 −4 0
ratio5
ln_val
1.0 0.5 0.0
−6 −4 −2 0
ratio5
log10_val
3 2 1 0
−2 −1 0
ratio5
log2_val
0.75 0.50 0.25 0.00
−7.5 −5.0 −2.5 0.0
requests
ln_val
0.15 0.10 0.05 0.00
4 8 12 16
requests
log10_val
0.4 0.3 0.2 0.1 0.0
2 4 6
requests
log2_val
0.10 0.05 0.00
5 10 15 20
viewability
ln_val
1.2 0.9 0.6 0.3 0.0
−7.5 −5.0 −2.5 0.0
viewability
log10_val
2 1 0
−3 −2 −1 0 1
viewability
log2_val
0.8 0.6 0.4 0.2 0.0
−8 −4 0
y
ln_val
0.3 0.2 0.1 0.0
−5 0
y
log10_val
0.6 0.4 0.2 0.0
−4 −2 0
y
log2_val
0.20 0.15 0.10 0.05 0.00
−10 −5 0 5

1.3 References