Mercari Price Suggestion Challenge

Ali Ezzat, Mohammed Ali

December 30, 2017

Contents

About

Mercari's challenge is to build an algorithm that automatically suggests the right product prices. You'll be provided user-inputted text descriptions of their products, including details like product category name, brand name, and item condition.

Exploratory Data Analysis

Data overview

- Load training data

```
train <- fread("Mercari-Price-Suggestion-Challenge/data/train.tsv", sep = "\t",
    stringsAsFactors = FALSE, showProgress = FALSE)
```

- Inspect structure

```
glimpse(train)
```

```
## Observations: 1,482,535
## Variables: 8
## $ train_id          <int> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13...
## $ name              <chr> "MLB Cincinnati Reds T Shirt Size XL", "Raze...
## $ item_condition_id <int> 3, 3, 1, 1, 1, 3, 3, 3, 3, 3, 2, 1, 2, 1, 3,...
## $ category_name     <chr> "Men/Tops/T-shirts", "Electronics/Computers ...
## $ brand_name        <chr> "", "Razer", "Target", "", "", "", "Acacia S...
## $ price             <dbl> 10, 52, 10, 35, 44, 59, 64, 6, 19, 8, 8, 34,...
## $ shipping          <int> 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0,...
```

```
## $ item_description  <chr> "No description yet", "This keyboard is in g...
```

At the first glance, excluding the *train_id* column, the numeric data are only 3 columns one of them is the response variable *price*, while the other two are factor data which mean that we have a very limited number of features and that mean we will need the help of the other four text predictors to build our model. So let us investigate these features one by one to see what we can do.

Price (The response/target variable)

Let's start with an analysis of the response0 variable: price. First, the range of item prices:

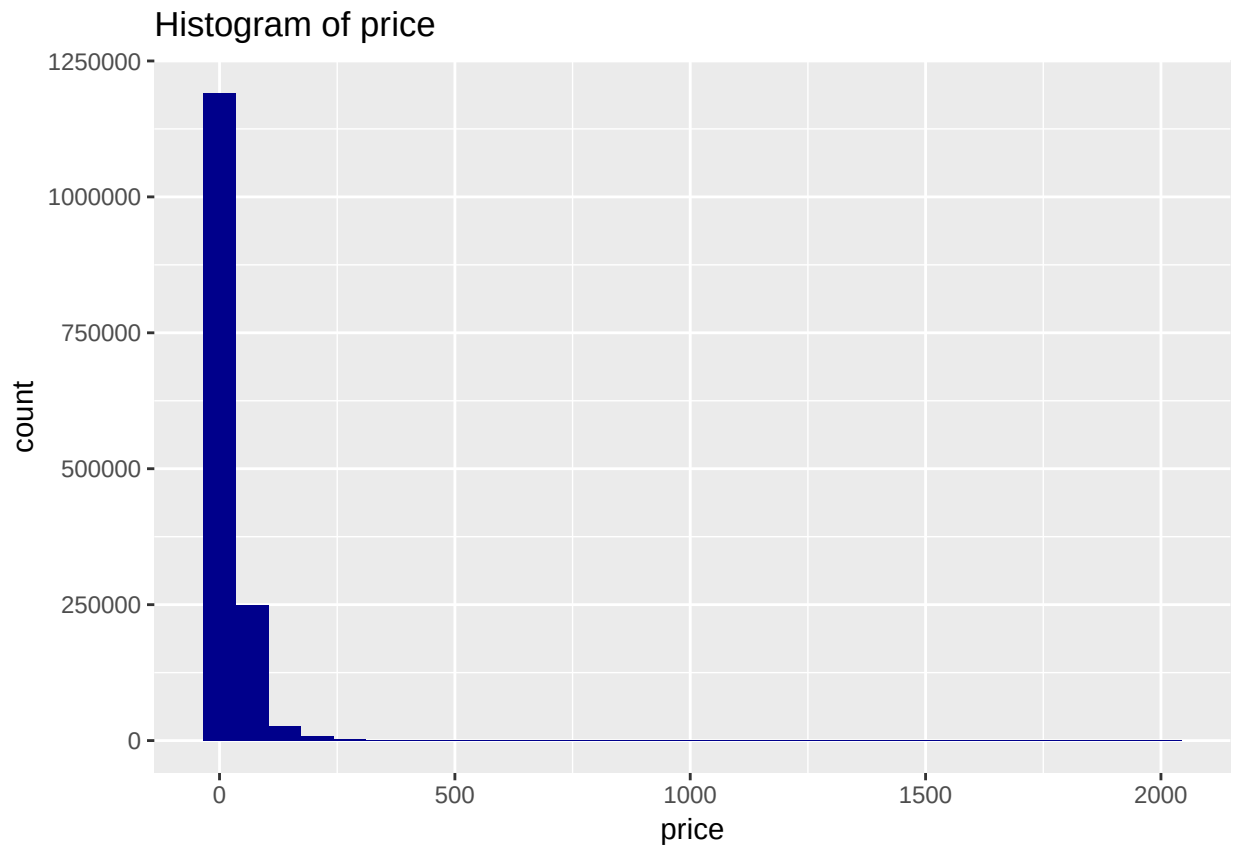
```
summary(train$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   10.00   17.00   26.74   29.00 2009.00
```

It seems there are items are givem as gifts (min. is 0), and we have a few very expensive items as well. Let us visualize it for more clear picture

```
ggplot(data = train, aes(x = price)) +
    geom_histogram(fill = 'darkblue') +
    labs(title = 'Histogram of price')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

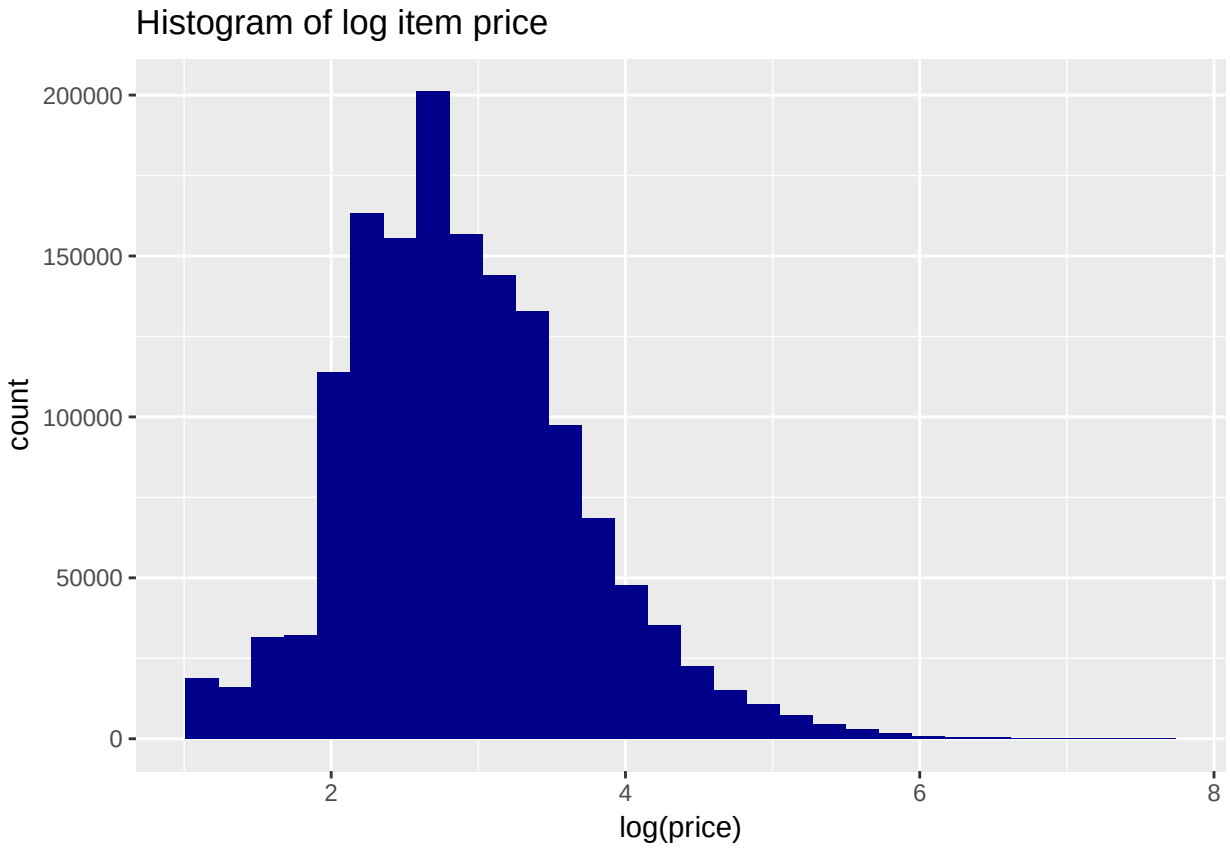


It seems because we have a very expensive items (and very view) we will need to take the log for better analysis

```
ggplot(data = train, aes(x = log(price))) +
    geom_histogram(fill = 'darkblue') +
    labs(title = 'Histogram of log item price')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 874 rows containing non-finite values (stat_bin).
```

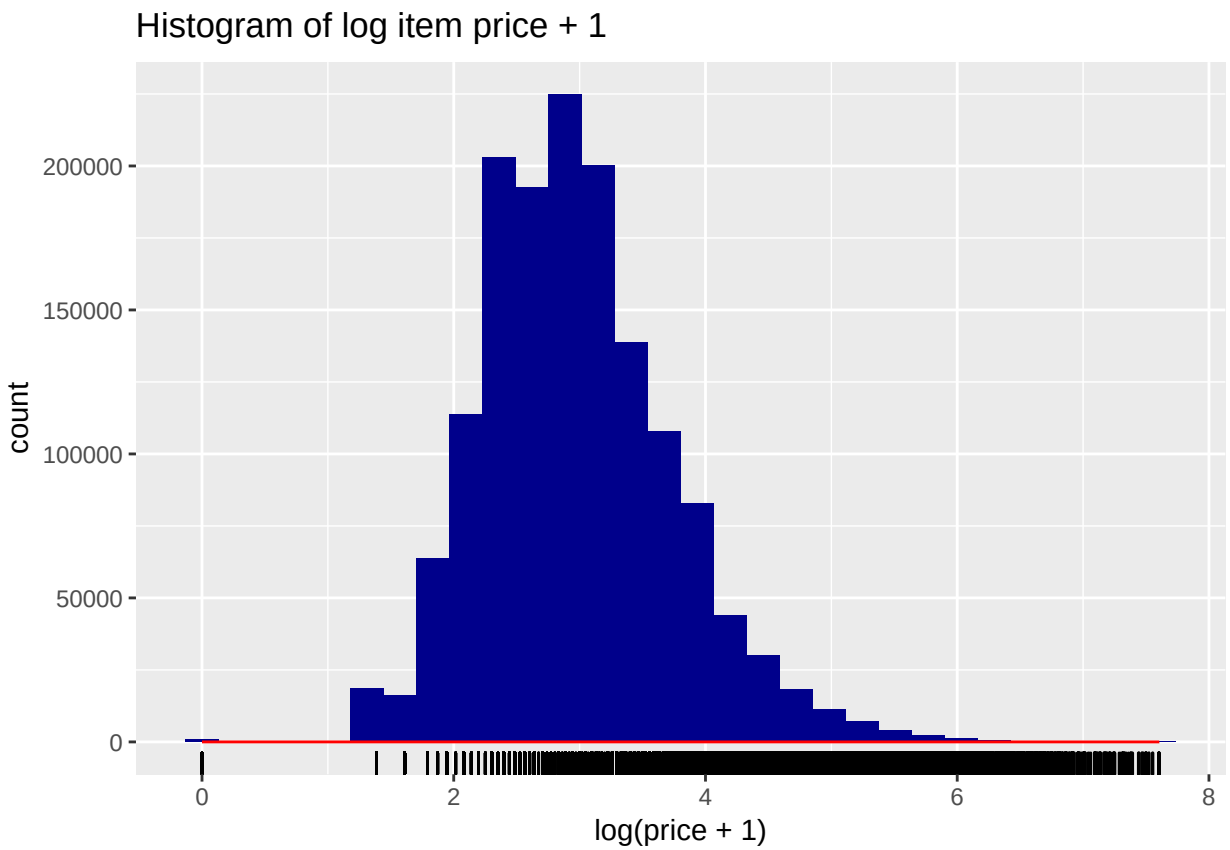


Now, it is much better and clear, but it seems taking log of prices made the free/gift items to be omitted, as taking log for 0 is undefined, so we will have to add a dummy 1 to include it with us in the plot, consider it a tax :P .

```
ggplot(data = train, aes(x = log(price + 1))) +
    geom_histogram(fill = 'darkblue') +
    labs(title = 'Histogram of log item price + 1') +
  geom_rug() +
  stat_function(fun = dnorm, colour = "red",
arg = list(mean = mean(train$price),
sd = sd(train$price)))
```

```
## Warning: Ignoring unknown parameters: arg
```

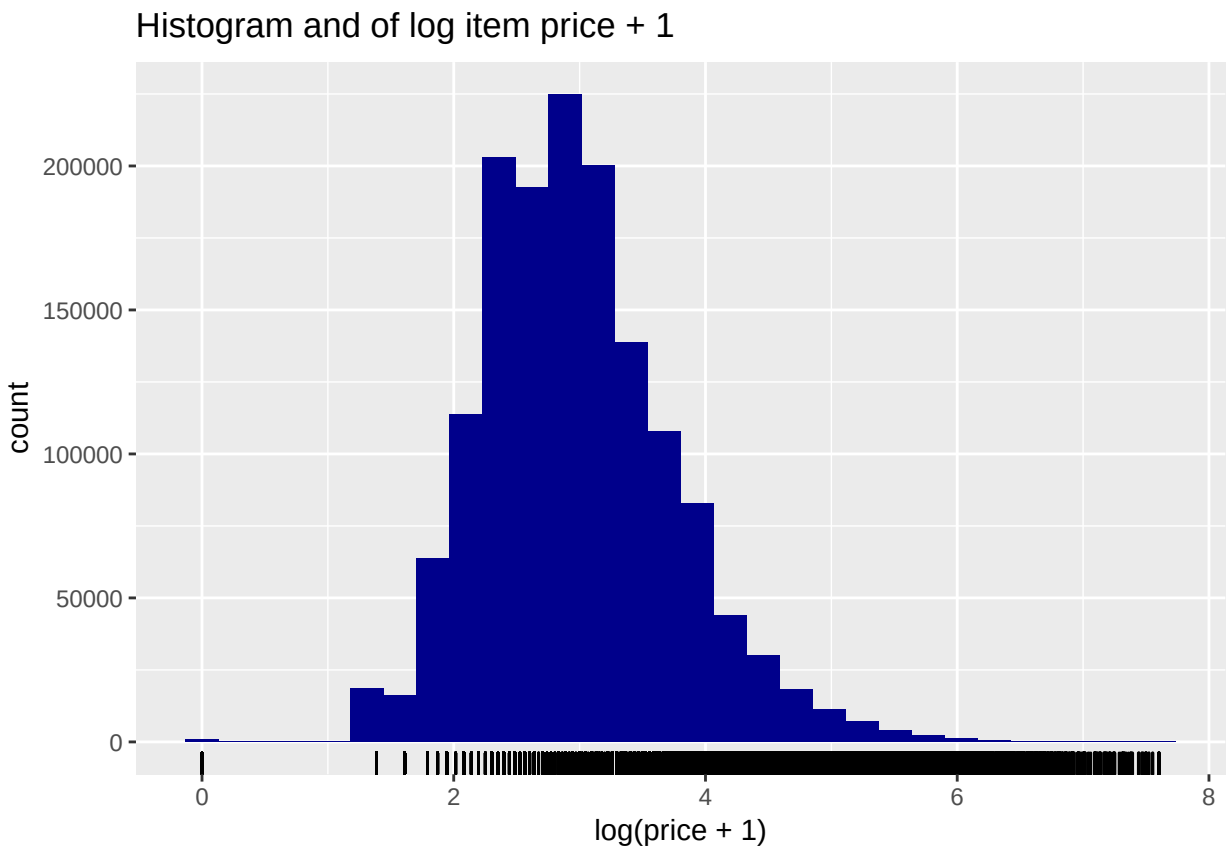
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



So, finally we have the gift/free items in the plot and it seems like an outlier along with the pricy data, as we have a nearly normal distibution for the prices. Let us investigate the distribution more.

```
ggplot(data = train, aes(x = log(price + 1))) +
    geom_histogram(fill = 'darkblue') +
    labs(title = 'Histogram and of log item price + 1') +
  geom_rug()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



It seems the data is centered between 3 and 7, let us investigate the statistics more

```
summary(log(train$price + 1))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.398   2.890   2.979   3.401   7.606
```

That confirms our induction from the plot and conclude our investigation, for now, around the price.

Item Condition

Item condition is a factor data, so let us convert it to a factor first.right format

```
#item condition factor
train$item_condition_id <- as.factor(train$item_condition_id)
levels(train$item_condition_id)
```

```
## [1] "1" "2" "3" "4" "5"
```

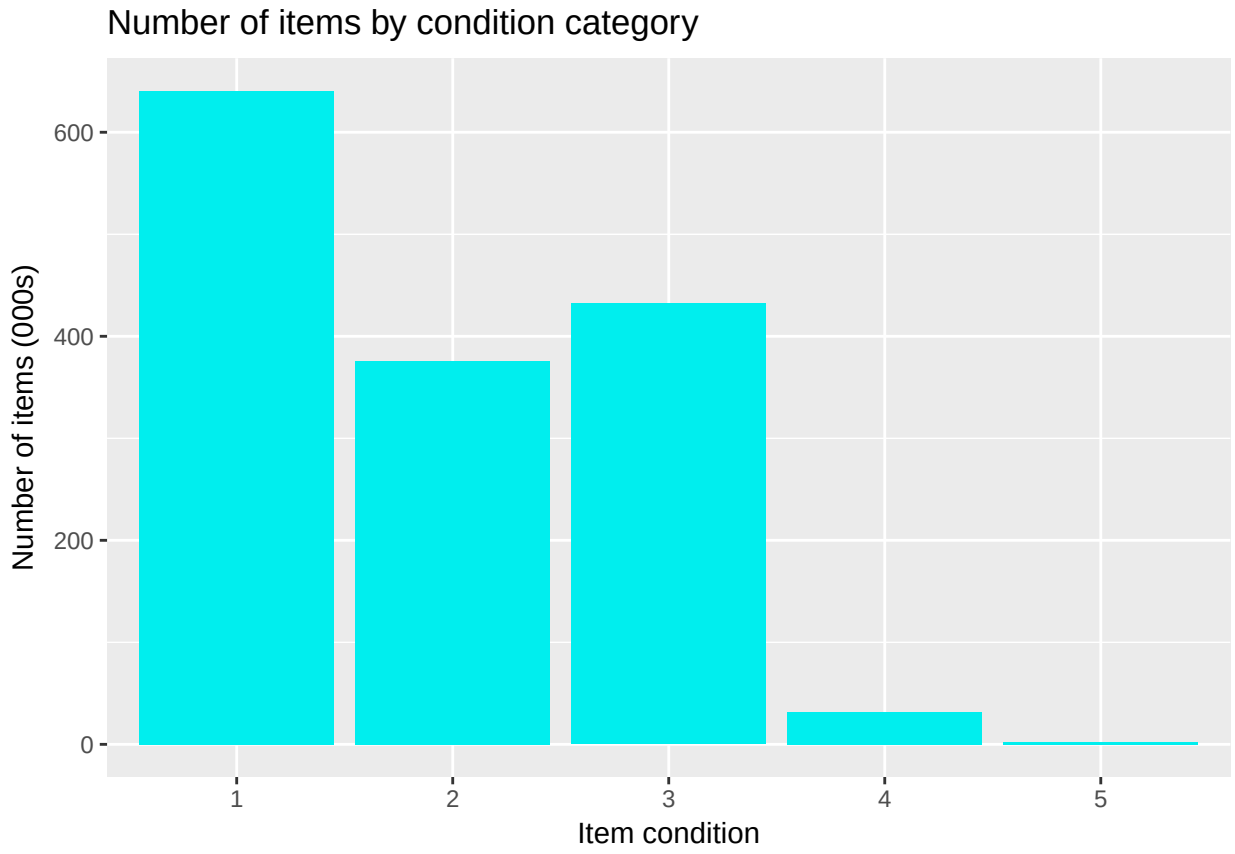
Now, let us see the statistics summary of items conditions

```
table(train$item_condition_id)
```

```
##
##      1      2      3      4      5
## 640549 375479 432161  31962   2384
```

We can confirm it more by the following plot

```
train[, .N, by = item_condition_id] %>%
    ggplot(aes(x = item_condition_id, y = N/1000)) +
    geom_bar(stat = 'identity', fill = 'cyan2') +
    labs(x = 'Item condition', y = 'Number of items (000s)', title = 'Number of items by condition categ
```



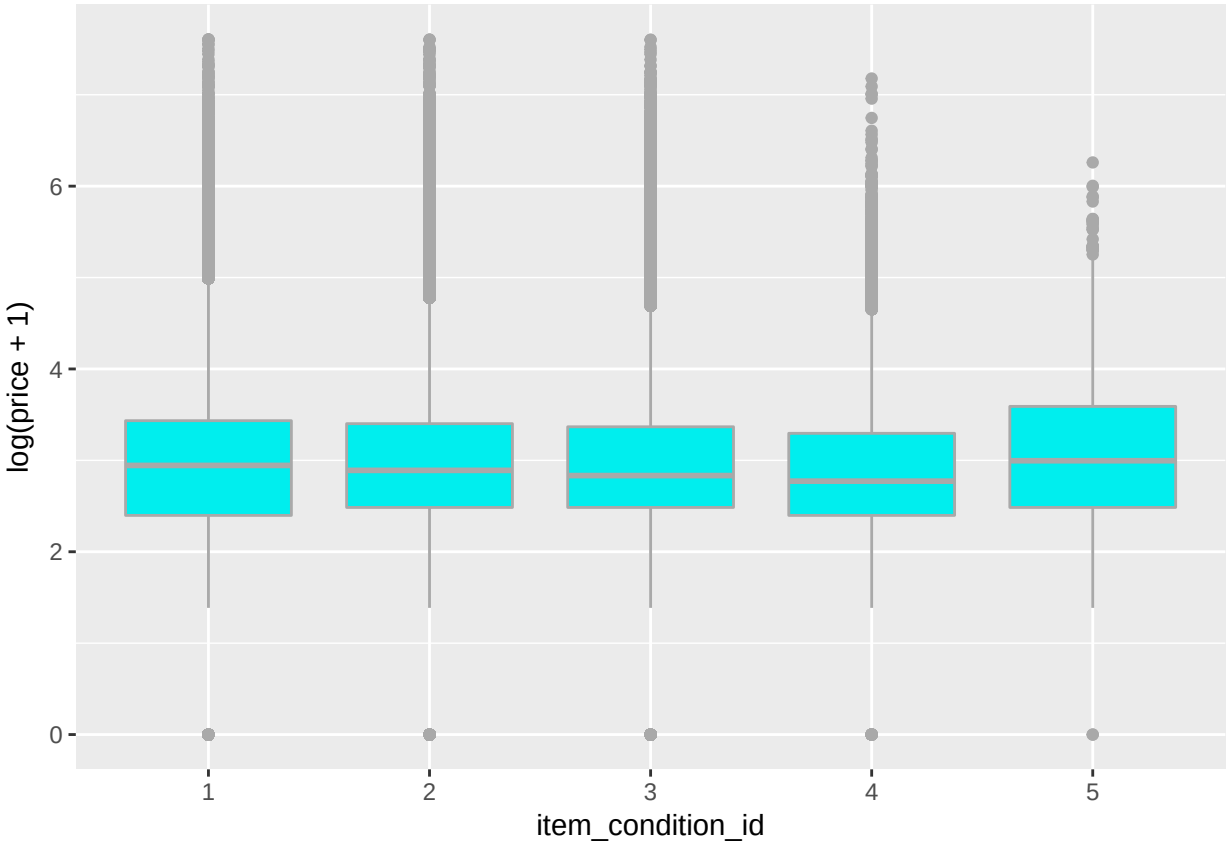
It seems most of the items are in condition *1* which is, I do not know there is no ordinal description in the compition. So we do not know if it *1* is the best or the worst. However, hanks to kaggler @Juraj for pointing out that in fact condition 1 is the best and 5 is the worst.

Now, let us compare the *item conditions* predictor aginst the response variable *price*

```
train[, .(.N, median_price = median(price)), by = item_condition_id][order(item_condition_id)]
```

```
##    item_condition_id      N median_price
## 1:                1 640549           18
## 2:                2 375479           17
## 3:                3 432161           16
## 4:                4  31962           15
## 5:                5   2384           19
```

```
ggplot(data = train, aes(x = item_condition_id, y = log(price + 1))) +
    geom_boxplot(fill = 'cyan2', color = 'darkgrey')
```



It seems that item condition is not the main contributor to the price as the best price at condition *5* with few items and the second best price at condition *1* with a lot of items.

Shipping

```
#shupping
train$shipping <- as.factor(train$shipping)
levels(train$shipping)
```

```
## [1] "0" "1"
```

```
table(train$shipping)
```

```
##
##      0      1
## 819435 663100
```

- Price summary

Model Building

Conclusion

Credit

Troy Walters

eBay acronyms

A-E B&W: black and white

BC: back cover (usually used as a description for books)

BIN: Buy It Now

CIP: customer initiated payment

DOA: dead on arrival (an item that doesn't work or is broken when it's received)

DSR: detailed seller rating (additional Feedback ratings buyers can give sellers)

EST: Eastern Standard Time

EUC: excellent used condition

F-I FAQ: frequently asked questions (a list of questions with answers.)

FB: Feedback

FC: fine condition

FOB: freight on board (usually means something has shipped)

FS: full screen (usually applied to a DVD or video format)

FVF: final value fee

G: good condition

GBP: Great Britain pounds

GU: gently used (item that has been used but shows little wear, accompanied by explanation of wear)

HP: home page

HTF: hard to find

HTML: hypertext markup language (the language used to create web pages)

IE: Internet Explorer

IM: instant messaging

INIT: initials

ISP: Internet service provider (a company that gives you access to the Internet)

J-M JPG: JPEG (preferred file format for pictures on eBay, pronounced "jay-peg")

LTBX: letterbox (video format that recreates a widescreen image)

LTD: limited edition

MNT: mint or in perfect condition (a subjective term that doesn't necessarily mean new)

MIB: mint in box

MIJ: made in Japan

MIMB: mint in mint box

MIMP: mint in mint package

MIP: mint in package

MNB: mint no box

MOC: mint on card

MOMC: mint on mint card

MONMC: mint on near mint card

MWBT: mint with both tags

MWMT: mint with mint tags

N-P NARU: not a registered user (also a suspended user)

NBW: never been worn

NC: no cover

NIB: new in box

NM: near mint

NOS: new old stock

NR: no reserve price (for an auction-style listing)

NRFB: never removed from box

NWT: new with tags

NWOB: new without box

NWOT: new without original tags

OEM: original equipment manufacturer

OOP: out of print

PST: Pacific Standard Time

Q-Z RET: retired

SCR: scratch

S/O: sold out

Sig: signature