

StatDescript

Yuxuan Li

9/2/2019

```
summary(bnb.data)

##           X          log_price      property_type
##   Min. :    1   Min. :0.000   Apartment :23737
##   1st Qu.:18602  1st Qu.:4.317   House     : 8049
##   Median :37011   Median :4.718   Condominium: 1298
##   Mean   :37081   Mean   :4.783   Townhouse  :  820
##   3rd Qu.:55712  3rd Qu.:5.220   Loft      :  616
##   Max.  :74111   Max.  :7.600   Other     :  289
##                   (Other)    : 1191
##           room_type      amenities      accommodates
##   Entire home/apt:20001  Length:36000   Min.   : 1.000
##   Private room     :14928   Class  :character  1st Qu.: 2.000
##   Shared room      : 1071   Mode   :character  Median  : 2.000
##                           Mean   : 3.155
##                           3rd Qu.: 4.000
##                           Max.  :16.000
##
##           bathrooms      bed_type      cancellation_policy
##   Min.  :0.00   Airbed       : 245   flexible     :10971
##   1st Qu.:1.00  Couch       : 136   moderate    : 9221
##   Median :1.00  Futon       : 392   strict      :15755
##   Mean   :1.24  Pull-out Sofa: 265   super_strict_30:   45
##   3rd Qu.:1.00  Real Bed    :34962   super_strict_60:    8
##   Max.  :8.00
##   NA's   :95
##           cleaning_fee      city      first_review      host_has_profile_pic
##   Mode  :logical    Boston : 1686   Min.   :2009-01-20   :  91
##   FALSE:9668     Chicago: 1800   1st Qu.:2015-06-27   f: 108
##   TRUE :26332    DC     : 2788   Median :2016-05-23   t:35801
##           LA     :10966   Mean   :2016-01-13
##           NYC    :15629   3rd Qu.:2017-01-02
##           SF     : 3131   Max.   :2017-10-03
##           NA's   :7717
##           host_identity_verified host_response_rate  host_since
##   :    91           100%   :20902   Min.   :2008-03-04
##   f:11828          : 8983   1st Qu.:2013-04-24
##   t:24081          90%   :1125   Median :2014-09-30
##           80%   : 564   Mean   :2014-07-21
##           0%    : 443   3rd Qu.:2015-12-22
##           50%   : 292   Max.   :2017-10-04
##           (Other): 3691   NA's   :91
##           instant_bookable last_review      latitude      longitude
##   f:26622          Min.   :2009-01-21   Min.   :33.34   Min.   :-122.51
##   t: 9378          1st Qu.:2017-01-06   1st Qu.:34.12   1st Qu.:-118.34
##           Median :2017-04-28   Median :40.66   Median : -77.00
##           Mean   :2017-03-12   Mean   :38.43   Mean   : -92.48
```

```

##            3rd Qu.:2017-09-08   3rd Qu.:40.75   3rd Qu.: -73.95
##            Max.    :2017-10-05   Max.    :42.39   Max.    : -70.99
##            NA's     :7696
##      neighbourhood   number_of_reviews review_scores_rating
##            : 3311   Min.    : 0.00   Min.    : 20.0
## Williamsburg       : 1349   1st Qu.: 1.00   1st Qu.: 92.0
## Bedford-Stuyvesant: 1091   Median  : 6.00   Median  : 96.0
## Bushwick          :  815   Mean    : 20.73   Mean    : 94.1
## Hollywood          :  674   3rd Qu.: 23.00   3rd Qu.:100.0
## Mid-Wilshire        :  673   Max.    :532.00   Max.    :100.0
## (Other)           :28087   NA's    :8143
##      zipcode        bedrooms      beds      price
## Length:36000      Min.    : 0.00   Min.    : 0.000   Min.    : 1.0
## Class :character  1st Qu.: 1.00   1st Qu.: 1.000   1st Qu.: 75.0
## Mode  :character  Median  : 1.00   Median  : 1.000   Median  :112.0
##               Mean  : 1.27   Mean    : 1.714   Mean    :161.1
##               3rd Qu.: 1.00   3rd Qu.: 2.000   3rd Qu.:185.0
##               Max.  :10.00   Max.    :16.000   Max.    :1999.0
##               NA's  :41     NA's    :59
##      post_wait      am_Internet      am_TV      am_AC
## Min.    :-931.0   Mode :logical   Mode :logical   Mode :logical
## 1st Qu.:  80.0    FALSE:1179     FALSE:9928     FALSE:9199
## Median  : 412.0   TRUE :34821    TRUE :26072    TRUE :26801
## Mean    : 581.5
## 3rd Qu.: 934.0
## Max.   :3204.0
## NA's   :7786
##      am_Kitchen      am_Heating      am_Washer      am_Dryer
## Mode  :logical   Mode :logical   Mode :logical   Mode :logical
## FALSE:3179       FALSE:3407       FALSE:14965     FALSE:15213
## TRUE  :32821      TRUE :32593      TRUE :21035     TRUE :20787
##
##
##
##
##      am_Parking      am_Shampoo      am_Essentials
## Mode  :logical   Mode :logical   Mode :logical
## FALSE:24594      FALSE:11926      FALSE:4901
## TRUE  :11406      TRUE :24074      TRUE :31099
##
##
##
##

```

number of amenity

First of first, take a look of numeric characters on price, including the numbers of **accomodates**, **beds**, **bedrooms**, and **bathrooms**. Do scatter plot and violin plot of (log-)price on each varaiable.

Conclusion: * The log price and those numbers have roughly positive relationship (sort of sigmoid). I think quadratic term might be necessary in linear regression. * Further controls are needed.

```

amen.data <- bnb.data %>%
  select(price, log_price, accommodates, beds, bedrooms, bathrooms) %>%

```

```

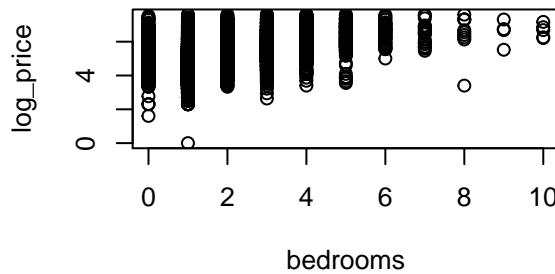
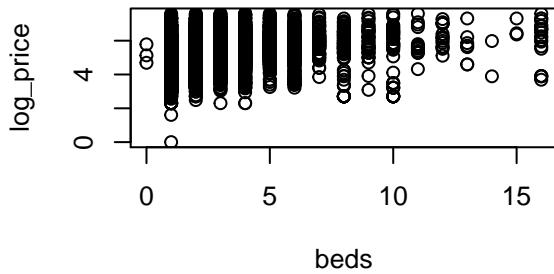
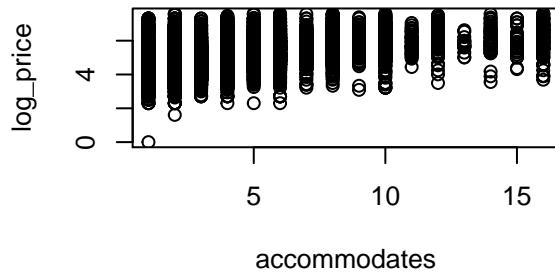
na.omit()

summary(amen.data)

##      price          log_price      accommodates      beds
##  Min.   : 1.0   Min.   :0.000   Min.   : 1.000   Min.   : 0.000
##  1st Qu.: 75.0  1st Qu.:4.317   1st Qu.: 2.000   1st Qu.: 1.000
##  Median : 113.0 Median :4.727   Median : 2.000   Median : 1.000
##  Mean   : 161.2 Mean  :4.784   Mean   : 3.159   Mean   : 1.715
##  3rd Qu.: 185.0 3rd Qu.:5.220   3rd Qu.: 4.000   3rd Qu.: 2.000
##  Max.   :1999.0  Max.   :7.600   Max.   :16.000   Max.   :16.000
##      bedrooms      bathrooms
##  Min.   : 0.00   Min.   :0.00
##  1st Qu.: 1.00   1st Qu.:1.00
##  Median : 1.00   Median :1.00
##  Mean   : 1.27   Mean   :1.24
##  3rd Qu.: 1.00   3rd Qu.:1.00
##  Max.   :10.00   Max.   :8.00

# log price to ammenity numbers
par(mfrow = c(2, 2))
for (i in 3:6){
  plot(amen.data[,2] ~ amen.data[,i],
       xlab = colnames(amen.data)[i], ylab = colnames(amen.data)[2])
}

```

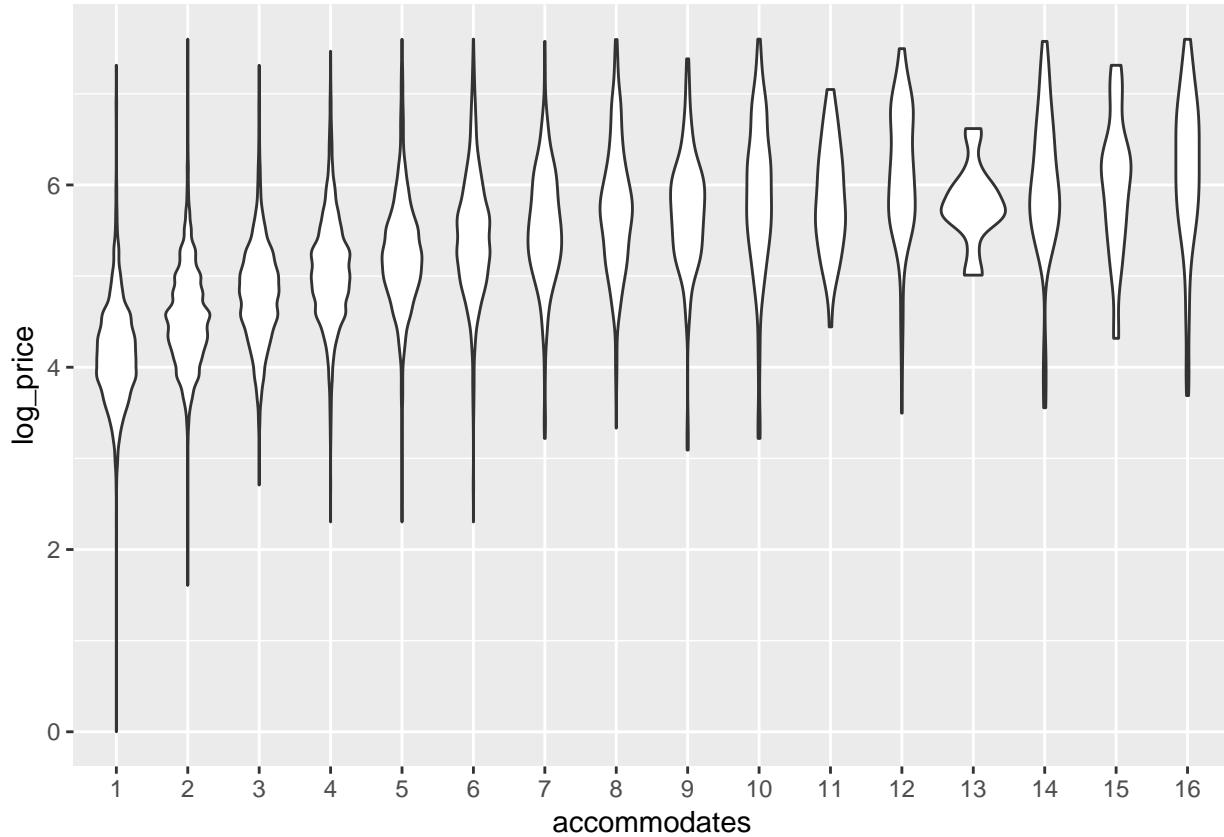


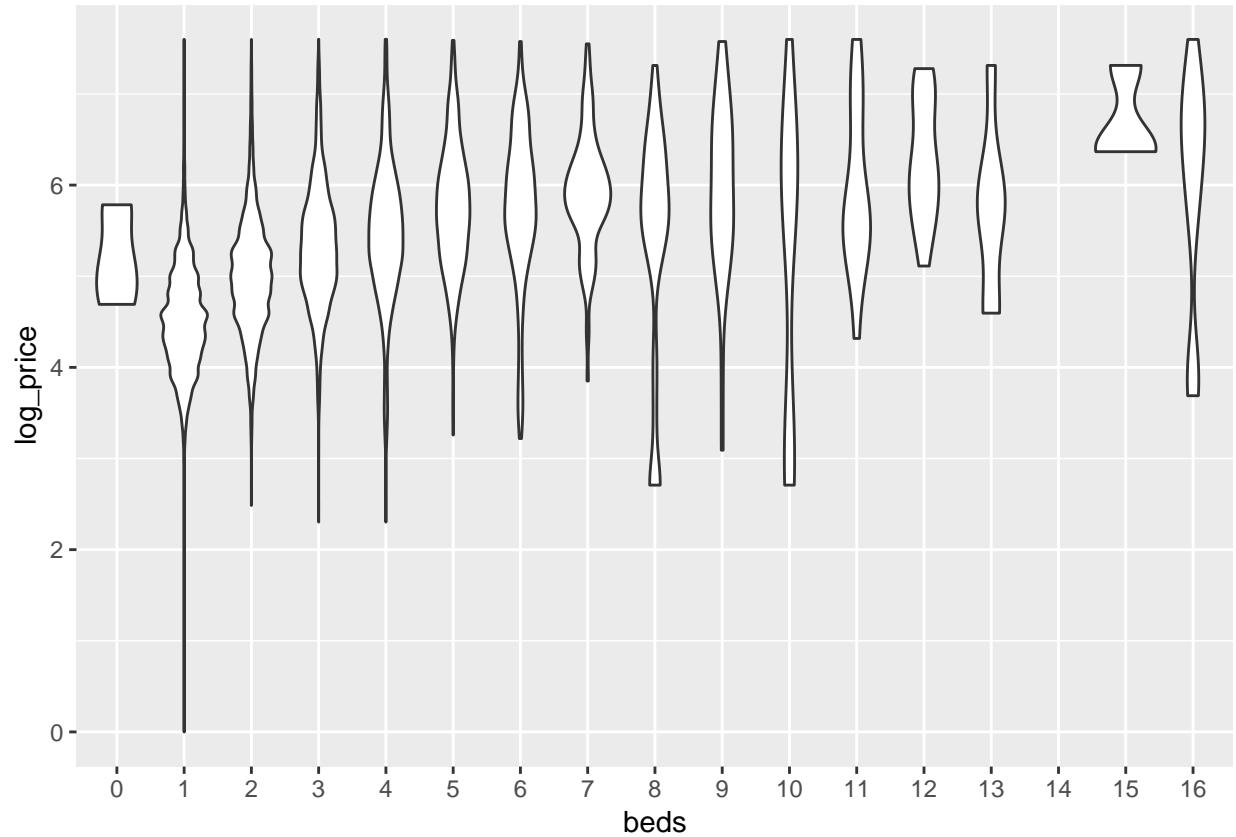
```

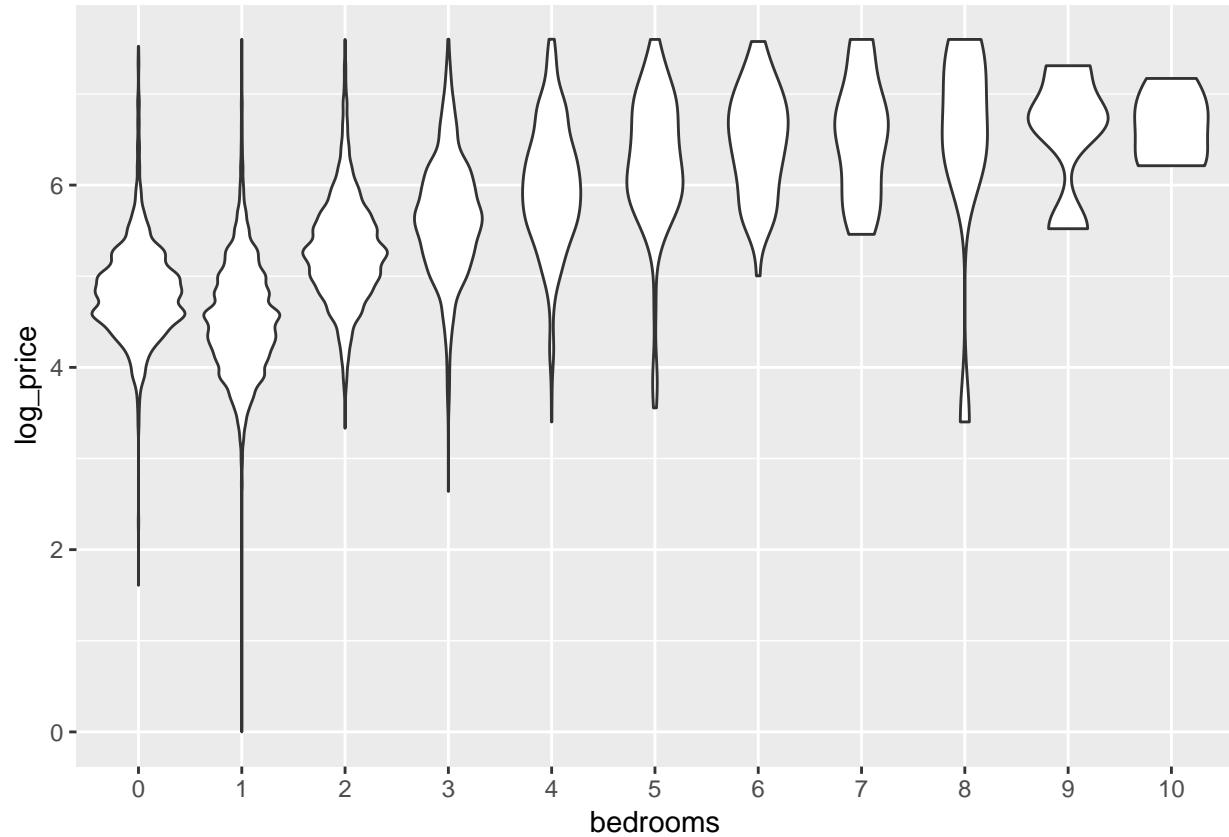
# better(?) version
for (i in 3:6){
  p <- ggplot(amen.data,aes(x = as.factor(amen.data[,i]), y = amen.data[,2])) +
    geom_violin(scale = "area") +
    xlab(colnames(amen.data)[i]) +
    ylab(colnames(amen.data)[2])

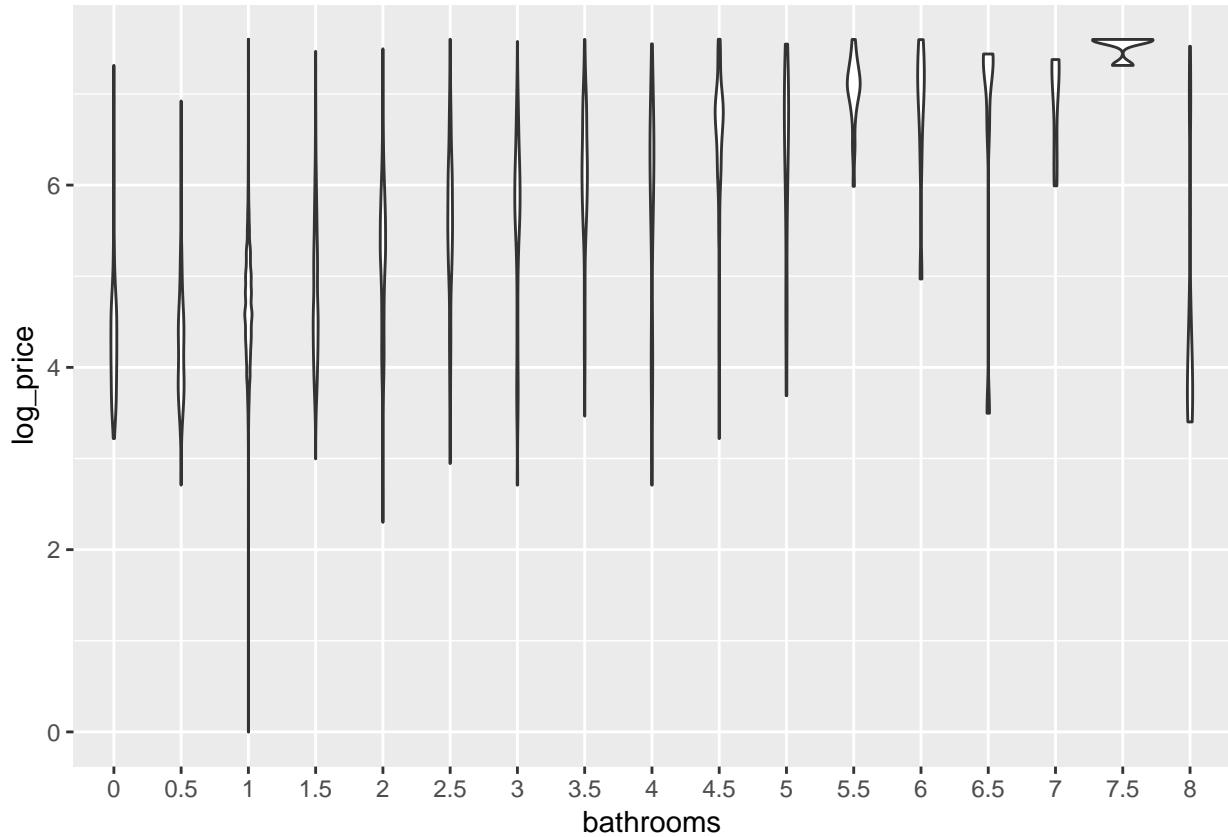
  print(p)
}

```

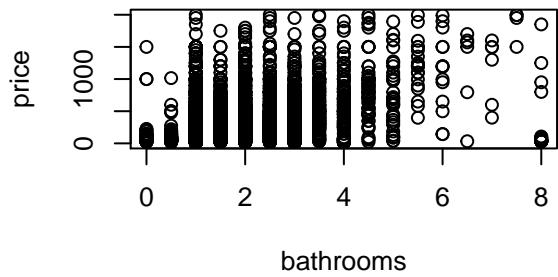
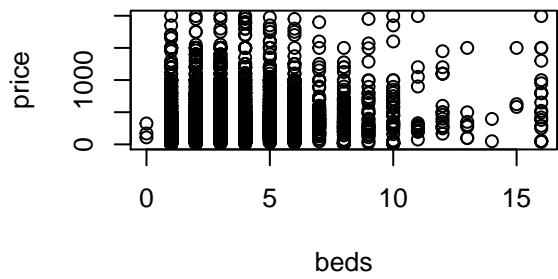
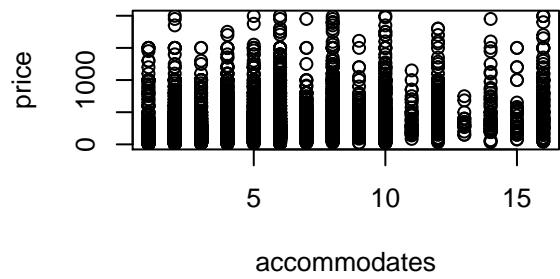








```
# price to ammenity numbers
par(mfrow = c(2, 2))
for (i in 3:6){
  plot(amen.data[,1] ~ amen.data[,i],
       xlab = colnames(amen.data)[i], ylab = colnames(amen.data)[1])
}
```



```
# better(?) version
for (i in 3:6){
  p <- ggplot(amen.data,aes(x = as.factor(amen.data[,i]), y = amen.data[,1])) +
    geom_violin(scale = "area") +
    xlab(colnames(amen.data)[i]) +
    ylab(colnames(amen.data)[1])

  print(p)
}
```

