

Information and Airbnb, An Analysis on the Impact of Reviews and Ratings on Airbnb Prices in Canada*

Alaina Hu and Ahad Qureshi

February 13, 2024

Table of contents

1	Introduction	1
2	Data	1

1 Introduction

2 Data

```
Rows: 87059 Columns: 9
-- Column specification -----
Delimiter: ","
chr (1): city
dbl (8): review, guest_satisfaction_overall, accuracy_rating, cleanliness_ra...
i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.

Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
i Please use `after_stat(density)` instead.
```

*Code and data are available at: https://github.com/alainahu/airbnb_analysis

```
Warning: Removed 102 rows containing non-finite values (`stat_bin()`).
```

```
Warning: Removed 2 rows containing missing values (`geom_bar()`).
```

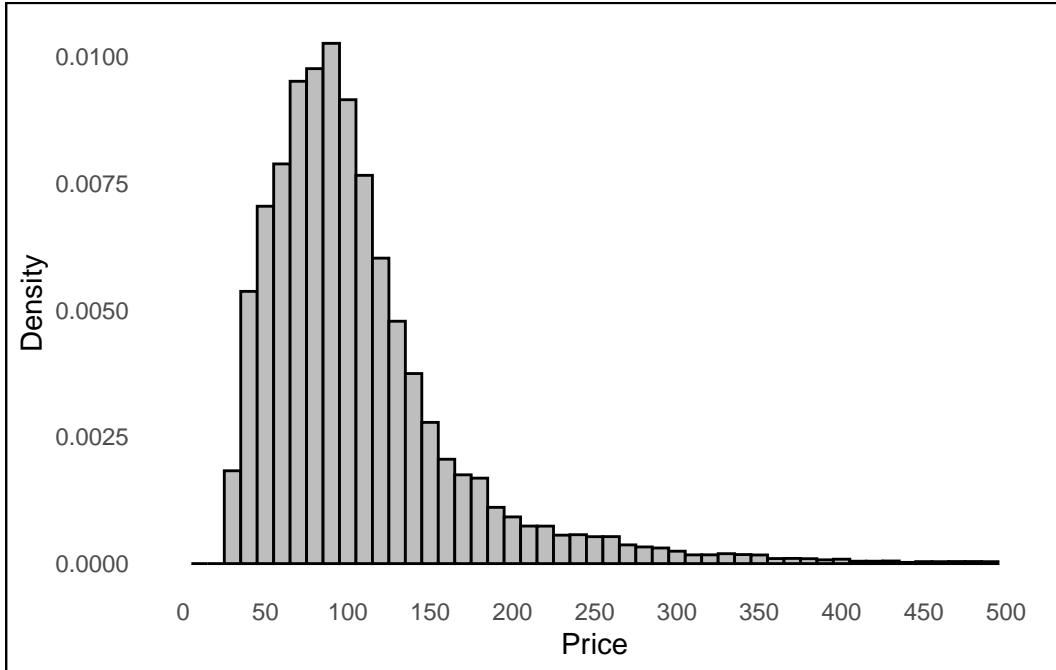


Figure 1: Distribution of Airbnb prices per night (including cleaning fees)

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
26.00	68.67	93.83	107.66	126.33	634.33

```
summary_df <- airbnb |>
  summarise(
    Mean_New_Price = mean(new_price, na.rm = TRUE),
    Median_New_Price = median(new_price, na.rm = TRUE),
    SD_New_Price = sd(new_price, na.rm = TRUE))
```

Table 1: Summary Statistics for the Reviews and Ratings

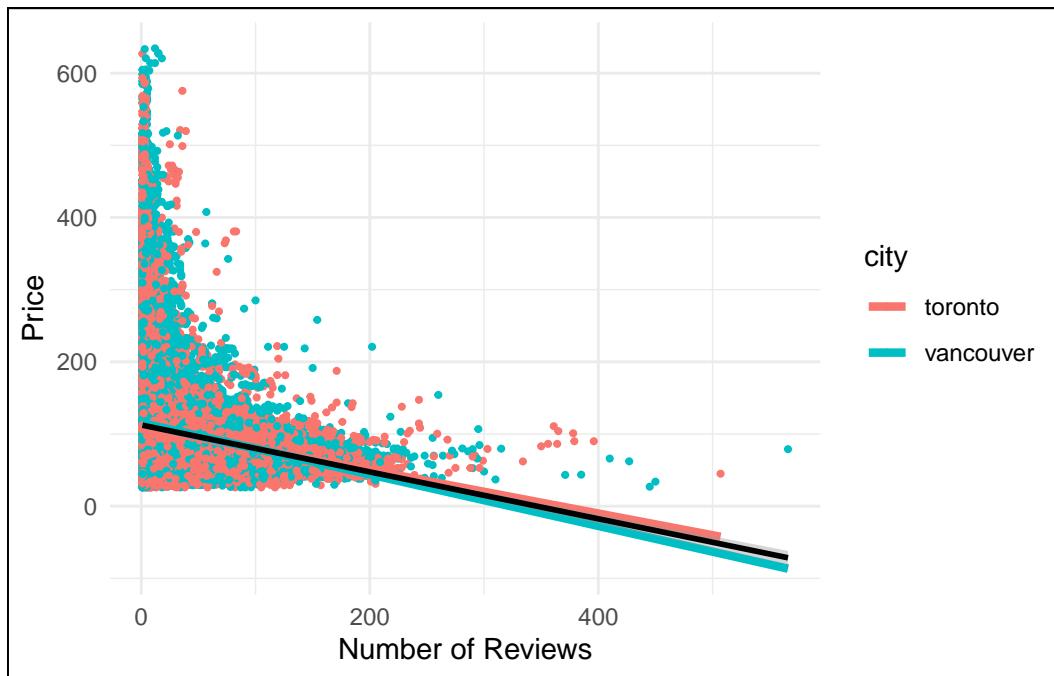
Variable	Mean	Median	Standard Deviation	Min	Max
Accuracy Rating	9.47	10	0.98	0	10
Cleanliness Rating	9.28	10	1.14	0	10

Variable	Mean	Median	Standard Deviation	Min	Max
Guest Overall Satisfaction	93.47	95	7.78	20	100
Location Rating	9.48	10	0.95	0	10
Number of Reviews	14.90	7	23.86	1	566
Value Rating	9.29	9	1.00	0	10

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
 i Please use `linewidth` instead.

```
`geom_smooth()` using formula = 'y ~ x'  

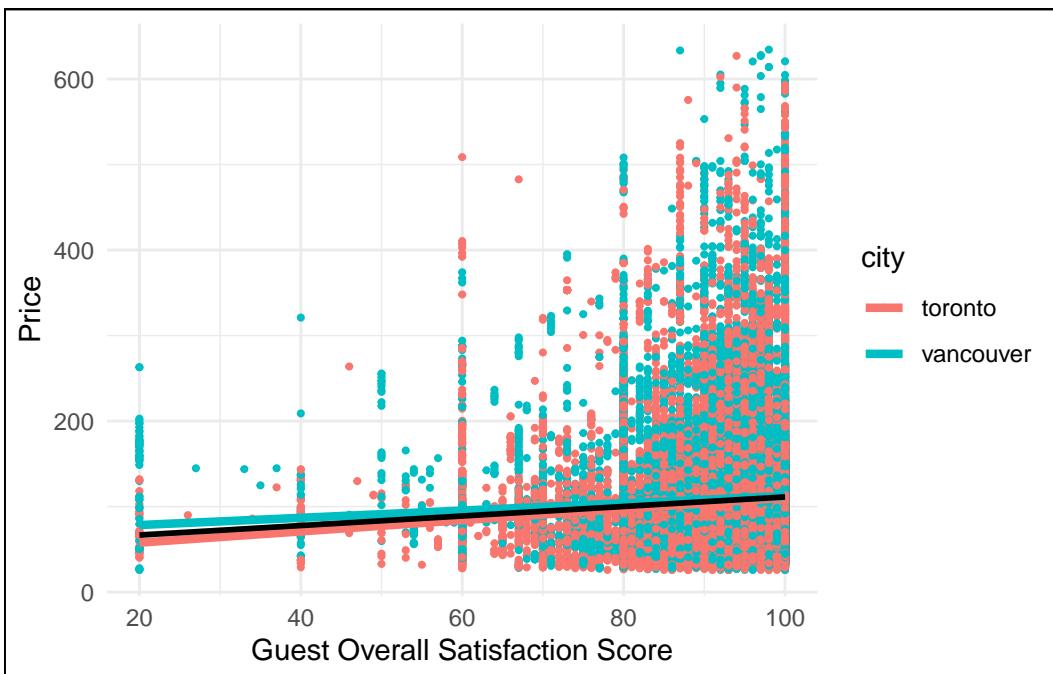
`geom_smooth()` using formula = 'y ~ x'
```



```
<ggproto object: Class ScaleDiscrete, Scale, gg>
  aesthetics: colour
  axis_order: function
  break_info: function
  break_positions: function
  breaks: waiver
  call: call
  clone: function
```

```
dimension: function
drop: TRUE
expand: waiver
get_breaks: function
get_breaks_minor: function
get_labels: function
get_limits: function
guide: legend
is_discrete: function
is_empty: function
labels: waiver
limits: NULL
make_sec_title: function
make_title: function
map: function
map_df: function
n.breaks.cache: NULL
na.translate: TRUE
na.value: NA
name: waiver
palette: function
palette.cache: NULL
position: left
range: environment
rescale: function
reset: function
scale_name: brewer
train: function
train_df: function
transform: function
transform_df: function
super: <ggproto object: Class ScaleDiscrete, Scale, gg>

`geom_smooth()` using formula = 'y ~ x'
`geom_smooth()` using formula = 'y ~ x'
```



```
<ggproto object: Class ScaleDiscrete, Scale, gg>
  aesthetics: colour
  axis_order: function
  break_info: function
  break_positions: function
  breaks: waiver
  call: call
  clone: function
  dimension: function
  drop: TRUE
  expand: waiver
  get_breaks: function
  get_breaks_minor: function
  get_labels: function
  get_limits: function
  guide: legend
  is_discrete: function
  is_empty: function
  labels: waiver
  limits: NULL
  make_sec_title: function
  make_title: function
  map: function
```

```
map_df: function
n.breaks.cache: NULL
na.translate: TRUE
na.value: NA
name: waiver
palette: function
palette.cache: NULL
position: left
range: environment
rescale: function
reset: function
scale_name: brewer
train: function
train_df: function
transform: function
transform_df: function
super:  <ggproto object: Class ScaleDiscrete, Scale, gg>
```