

楽天市場



22229552 LOEDSINAUDOM **JUTHATHIP**

22229568 PRAJEEYACHAT **ZACHARIAH**

20224524 Lee **Sherman**

Rakuten Ichiba

One of the biggest Japanese e-commerce site.



A screenshot of the Rakuten Ichiba mobile website. At the top, there is a navigation bar with links like 'DEAL', 'Fashion', 'Travel', 'Card', 'Stock', 'Bank', 'Books', 'E-books', 'Lacma', 'Rebate', and 'Music'. Below the navigation is a search bar labeled 'ワード検索'. A prominent banner at the top right says 'Rakuten Mobile スマホそのまま乗り換え' and '他社から電話番号そのまま乗り換え & 初めてお申し込みで 増量中 10,000'. The main content area features a 'PREMIUM DAYS' section with a yellow button-down shirt and a plaid tie, advertising a 10x points multiplier for entry. Other banners include 'Rakuten Fashion' with 'THE NORTH FACE', 'RALPH LAUREN', 'marimekko', and 'SK-II and more'. There are also sections for 'Rakuten Turbo' and 'Rakuten Ichiba English guide'. A bottom banner offers a 3x points multiplier for shopping at multiple stores. A note at the bottom left states: '①ポイント未利用時の倍率。上限・条件あり。反映タイミングは'.

Rakuten Public API

Utilizing Rakuten Ichiba Ranking API



The screenshot shows the Rakuten Developers API documentation. At the top right, there are links for "Sign in | 日本語", "+ New App", and "Your". The main navigation bar includes "Rakuten Developers", "INTRODUCTION", "API", "TOOL BOX", "BLOG", and "HELP". Below the navigation, a breadcrumb trail shows "Home > API". The main content area is titled "API" and contains several sections:

- Rakuten Ichiba API**
 - Rakuten Ichiba Item Search API
 - Rakuten Ichiba Genre Search API
 - Rakuten Ichiba Tag Search API
 - Rakuten Ichiba Ranking API
 - Rakuten Product Search API
- Rakuten Books APIs**
 - Books Total Search API
 - Books Book Search API
 - Books CD Search API
 - Books DVD/Blu-ray Search API
 - Books Foreign Search API
 - Books Magazine Search API
 - Books Game Search API
 - Books Software Search API
 - Books Genre Search API
- Rakuten Travel APIs**
 - Simple Hotel Search API

On the right side, there is a table titled "Rakuten Ichiba APIs" with columns for "API name", "Description", "Affiliate support", "API auth type", and "Test". The table lists three APIs with their descriptions and approval requirements:

API name	Description	Affiliate support	API auth type	Test
Rakuten Ichiba Item Search API	Rakuten Ichiba Item Search API enables developers to retrieve information of selling items on Rakuten Ichiba excluding those listed on auctions, flea markets, customer to customer auctions (C2C) or joint purchases on Rakuten Ichiba. Developers are allowed to obtain the item information by specifying Keyword as well as to filter the search with Item Code, Shop Name or Genre Name.		App ID approval	Test
Rakuten Ichiba Genre Search API	Rakuten Ichiba Genre Search API enables developers to retrieve genre name and genre structure on Rakuten Ichiba. Developers are allowed to obtain the genre information by specifying Genre ID.		App ID approval	Test
Rakuten Ichiba Tag Search API	Rakuten Ichiba Tag Search API enables developers to retrieve tag name and tag structure on Rakuten Ichiba. Developers are allowed to obtain the tag information by specifying Tag ID that can be retrieved with either Rakuten Ichiba Item Search API or Rakuten Ichiba Genre Search API.		App ID approval	Test

最近チェックしたジャンル

ヘッドホン・イヤホン
石けん・ボディソープ
柔軟剤
シャンプー
ボディローション・ミルク

Daily ジャンル一覧

レディースファッション	□
メンズファッション	□
靴	□
バッグ・小物・ブランド雑貨	□
下着・ナイトウェア	□
ジュエリー・アクセサリー	□
腕時計	□
食品	□
スイーツ・お菓子	□
水・ソフトドリンク	□
ビール・洋酒	□
日本酒・焼酎	□
スマートフォン・タブレット	□

価格や送料は、商品のサイズや色によって異なる場合があります。その他の注意は[こちら](#)

対象商品 200,000,000 点以上！ 毎日更新!!

総合ランキング

2025年10月16日(木)更新

デイリー

リアルタイム



【最大100%ポイントバック!!】 【販売開始★】福袋おせち◆高級おせちを含む54種のおせちのいずれかをお届け

【送料無料...】

↑ 海鮮かに処

39ショップ

★★★★★ レビュー(38,230件)

★ お気に入り

12,345円

1位



【特典】Pokemon LEGENDS Z-A(【早期購入特典】サーナイトナイトを持ったラルトス)

↑ 楽天ブックス

39ショップ

★★★★★ レビュー(12件)

★ お気に入り

6,146円

2位

STAY



【特典】Pokemon LEGENDS Z-A Nintendo Switch 2 Edition(【早期購入特典】サーナイトナイトを持ったラルトス)

↑ 楽天ブックス

39ショップ

★★★★★ レビュー(1件)

★ お気に入り

7,009円

3位



楽天市場のイチオシ



総合1位に輝いたのは!?

年間ベストショップ

Rakuten SHOP OF THE YEAR

月間ベストショップ

Rakuten SHOP OF THE MONTH

地域ベストショップ

Rakuten SHOP OF THE AREA

売れ筋ランキング

全品10%以上ポイントバック

スーパーDEAL

ブランドファッション人気ランキング

Rakuten Fashion

楽天ふるさと納税ランキング

人気のお礼の品をチェック!

リアルタイムランキング

最新情報

Rakuten Market Ranking list as of 15 Oct

Setting up our project

```
1 import requests  
2 import json  
3 import pandas as pd  
4 from scipy import stats  
5 import numpy as np  
6 import matplotlib.pyplot as plt
```

```
1 params = {  
2     'applicationId': APP_ID,  
3     'format': 'json',  
4     'keyword': '-reviewCount'  
5 }
```

Necessary Libraries

From getting the data → Cleaning → Analyzing → Visualization

Setting up the query

We first get the `APP_ID` from Rakuten API page. We use the keyword to look for item with the most review count.

Getting the Data

Using **for loop** to go through 34 pages

```
● ○ ●
1 data = []
2 for i in range(1,35):
3     res = requests.get(f'https://app.rakuten.co.jp/services/api/IchibaItem/Ranking/20220601?page={i}', params=params).json()
4     # dropping: mediumImageUrls / smallImageUrls
5     for item in res['Items']:
6         if 'mediumImageUrls' in item['Item']:
7             del item['Item']['mediumImageUrls']
8         if 'smallImageUrls' in item['Item']:
9             del item['Item']['smallImageUrls']
10        data.append(item['Item'])
11 data
```

```
● ○ ●
1 len(data)
2 >>>997
```

The data

It would look something like this

```
1  {'affiliateRate': '2.0',
2  'affiliateUrl': '',
3  'asurakuArea': '',
4  'asurakuClosingTime': '',
5  'asurakuFlag': 0,
6  'availability': 1,
7  'carrier': 0,
8  'catchcopy': '【楽天ブックスならいつでも送料無料】',
9  'creditCardFlag': 1,
10 'endTime': '',
11 'genreId': '566404',
12 'hasPriceRange': 0,
13 'imageFlag': 1,
14 'itemCaption': '任天堂 Nintendo Switchピカチュウ チコリータ ポカブ ワニノコ ポケモンZA',
15 'itemCode': 'book:21683049',
16 'itemName': '【特典】Pokemon LEGENDS Z-A(【早期購入特典】サーナイトナイトを持ったラルトス',
17 'itemPrice': '6146',
18 'itemPriceBaseField': 'item_price_min3',
19 'itemPriceMax1': '6146',
20 'itemPriceMax2': '6146',
```

Item Properties

Almost 40 properties was returned



```
1 data_keys = list(data[0].keys())
```



```
1 dict_keys(['affiliateRate', 'affiliateUrl', 'asurakuArea', 'asurakuClosingTime',
2             'asurakuFlag', 'availability', 'carrier', 'catchcopy',
3             'creditCardFlag', 'endTime', 'genreId', 'hasPriceRange',
4             'imageFlag', 'itemCaption', 'itemCode', 'itemName', 'itemPrice',
5             'itemPriceBaseField', 'itemPriceMax1', 'itemPriceMax2', 'itemPriceMax3',
6             'itemPriceMin1', 'itemPriceMin2', 'itemPriceMin3', 'itemUrl',
7             'pointRate', 'pointRateEndTime', 'pointRateStartTime', 'postageFlag',
8             'rank', 'reviewAverage', 'reviewCount', 'shipOverseasArea', 'shipOverseasFlag',
9             'shopAffiliateUrl', 'shopCode', 'shopName', 'shopOfTheYearFlag', 'shopUrl', 'startTime', 'taxFlag'])
```

Convert to Dictionary

Preparing for a dataframe

```
1 #creating a dictionary from data
2 new_dict = {}
3 for attribute in data_keys:
4     values = []
5     for item in data:
6         if attribute in item:
7             values.append(item[attribute])
8         else:
9             values.append('')
10    new_dict[attribute] = values
```

```
1 df = pd.DataFrame(new_dict)
```

	affiliateRate	affiliateUrl	asurakuArea	asurakuClosingTime	asurakuFlag	availability	carrier	catchcopy	creditCardFlag	endTime	...	reviewCount	shipOverseasArea	shipOverseasFlag	shopAffiliateUrl	shopCode	shopName	shopOfTheYearFlag	shopUrl
0	2.0				0	0	0	【楽天ブックスならいつでも送料無料】	1	...	263		0		book	楽天ブックス	0	https://www.rakuten.co.jp/book/?rafcid=wsc_i_r...	
1	2.0				0	0	0	【楽天ブックスならいつでも送料無料】	1	...	31		0		book	楽天ブックス	0	https://www.rakuten.co.jp/book/?rafcid=wsc_i_r...	
2	2.0				0	0	0	【楽天ブックスならいつでも送料無料】	1	...	26		0		book	楽天ブックス	0	https://www.rakuten.co.jp/book/?rafcid=wsc_i_r...	
3	2.0				0	0	0	【楽天ブックスならいつでも送料無料】	1	...	5		0		book	楽天ブックス	0	https://www.rakuten.co.jp/book/?rafcid=wsc_i_r...	
4	4.0				0	1	0	骨も塩分も気にせず家族みんなが安心！傷品や規格外サイズの訳あり商品	1	...	29048		0	yamagoiida	骨取り魚の飯田商店	1	https://www.rakuten.co.jp/yamagoiida/?rafcid=w...		

5 rows × 41 columns

The next step is to clean it

The Data Frame

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 997 entries, 0 to 996
Data columns (total 41 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   affiliateRate    997 non-null    object  
 1   affiliateUrl     997 non-null    object  
 2   asurakuArea       997 non-null    object  
 3   asurakuClosingTime 997 non-null    object  
 4   asurakuFlag       997 non-null    int64  
 5   availability      997 non-null    int64  
 6   carrier           997 non-null    int64  
 7   catchcopy          997 non-null    object  
 8   creditCardFlag    997 non-null    int64  
 9   endTime            997 non-null    object  
 10  genreId           997 non-null    object  
 11  hasPriceRange     997 non-null    int64  
 12  imageFlag          997 non-null    int64  
 13  itemCaption        997 non-null    object  
 14  itemCode           997 non-null    object  
 15  itemName           997 non-null    object  
 16  itemPrice          997 non-null    object  
 17  itemPriceBaseField 997 non-null    object  
 18  itemPriceMax1      997 non-null    object  
 19  itemPriceMax2      997 non-null    object  
 20  itemPriceMax3      997 non-null    object  
 21  itemPriceMin1      997 non-null    object  
 22  itemPriceMin2      997 non-null    object  
 23  itemPriceMin3      997 non-null    object  
 24  itemUrl            997 non-null    object  
 25  pointRate          997 non-null    int64  
 26  pointRateEndTime   997 non-null    object  
 27  pointRateStartTime 997 non-null    object  
 28  postageFlag         997 non-null    int64  
 29  rank               997 non-null    int64  
 30  reviewAverage      997 non-null    object  
 31  reviewCount         997 non-null    int64  
 32  shipOverseasArea   997 non-null    object  
 33  shipOverseasFlag   997 non-null    int64  
 34  shopAffiliateUrl   997 non-null    object  
 35  shopCode            997 non-null    object  
 36  shopName            997 non-null    object  
 37  shopOfTheYearFlag   997 non-null    int64  
 38  shopUrl             997 non-null    object  
 39  startTime           997 non-null    object  
 40  taxFlag             997 non-null    int64  
dtypes: int64(13), object(28)
memory usage: 319.5+ KB
```

```
1 float_col = ['itemPrice', 'itemPriceMax1', 'itemPriceMax2', 'itemPriceMax3',
 2                 'itemPriceMin1', 'itemPriceMin2', 'itemPriceMin3', 'reviewAverage', ]
```

```
1 # turn string number into actual float
2 df[float_col] = df[float_col].astype(float)
```

```
1 # drop columns with uniform values or is completely empty
2 df = df.drop(['carrier', 'shipOverseasFlag', 'taxFlag', 'affiliateUrl',
 3                  'imageFlag', 'asurakuArea', 'asurakuClosingTime', 'asurakuFlag',
 4                  'shopAffiliateUrl', 'creditCardFlag'], axis=1)
```

```
1 df.isnull().sum()
```

Research Topic

"How product display features (e.g., price, shipping fee) influence consumer engagement on Rakuten Ichiba."

Hypothesis Testing

Connect to 3 Hypotheses:

- Price range display → review presence ?
- Price range display → review positivity?
- Free shipping → review average?

キーワードから探す

買い物かご お知らせ myクーポン 閲覧履歴 お気に入り 購入履歴 ヘルプ / 不適切な商品を報告 この商品の関連商品 ▾ 楽天市場での本商品の評価を見る ポイントアカウント

10kg

備蓄米10kg
令和3年産 備蓄米 米 10kg 生活応援 こめ 備蓄

39ショップ

商品番号: r-0000

2,480~7,970円 送料無料

22~73ポイント(1倍)

楽天カード新規入会で最短2分で
2,000ポイントゲット!

【重要】
転売目的の購入禁止

より多くのお客様に備蓄米をお届けするため、
転売目的でのご購入は固くお断りいたします。ご理解とご協力をお願い申し上げます。

弊社が上記に該当、もしくは、複数回の購入の繰り返し、不正行為、利用規約違反、その他運営上の趣旨に反していると合理的に判断した場合、ご注文をキャンセルさせていただく場合がございます。

【重要】
ご購入に関する注意点

①注文後の変更
注文した後の住所変更・日時指定などは受け付けませんので、注文確定前に必ず再度ご確認ください。

②納期について
納期に関して、順次発送をしているため日

商品詳細を選択

サイズ: 未選択

5kg(5kg×1袋)
2,480円
× 売り切れ

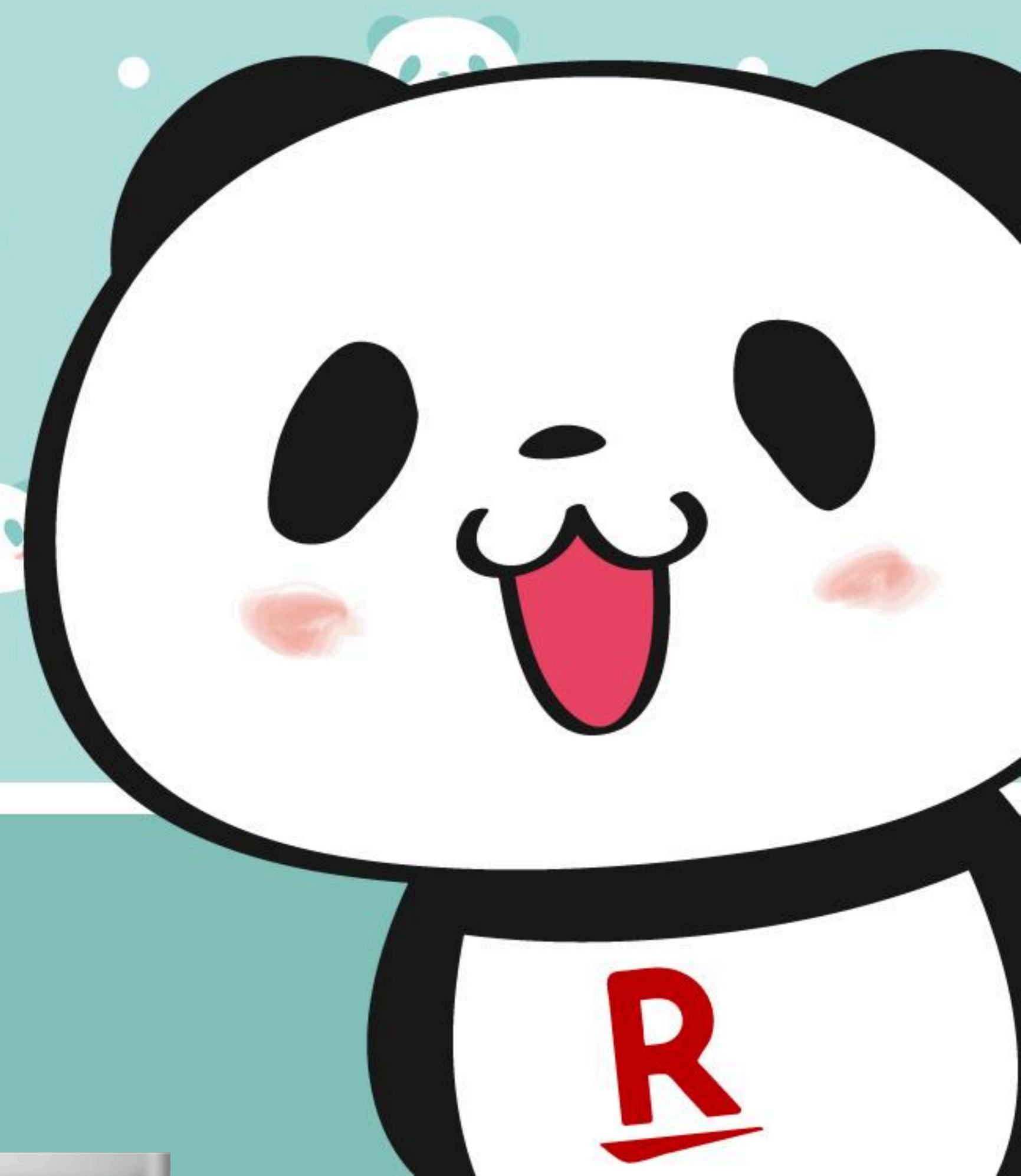
10kg(5kg×2袋)
4,880円
× 売り切れ

20kg(5kg×4袋)
7,970円
× 売り切れ

品質保持の観点から北海道・離島への配送は承っておりません。ご不便をおかけしますが、何卒ご理解のほどよろしくお願い致します。

選択してください

順次発送となるため、お届け日時の指定は不可となります。



一ワードから探す

買い物かご お知らせ myクーポン 閲覧履歴 お気に入り 購入履歴 楽天市場でのポイント

骨取りさば(2kg) | 骨なし骨抜きお弁当 送料無料 鯖魚 冷凍無添加 減塩 送料無料 アレンジ簡単ごはん 食品まとめ買い離乳食 お弁当 天然内祝 夏ギフト お中元 暑中お見舞い

39ショップ

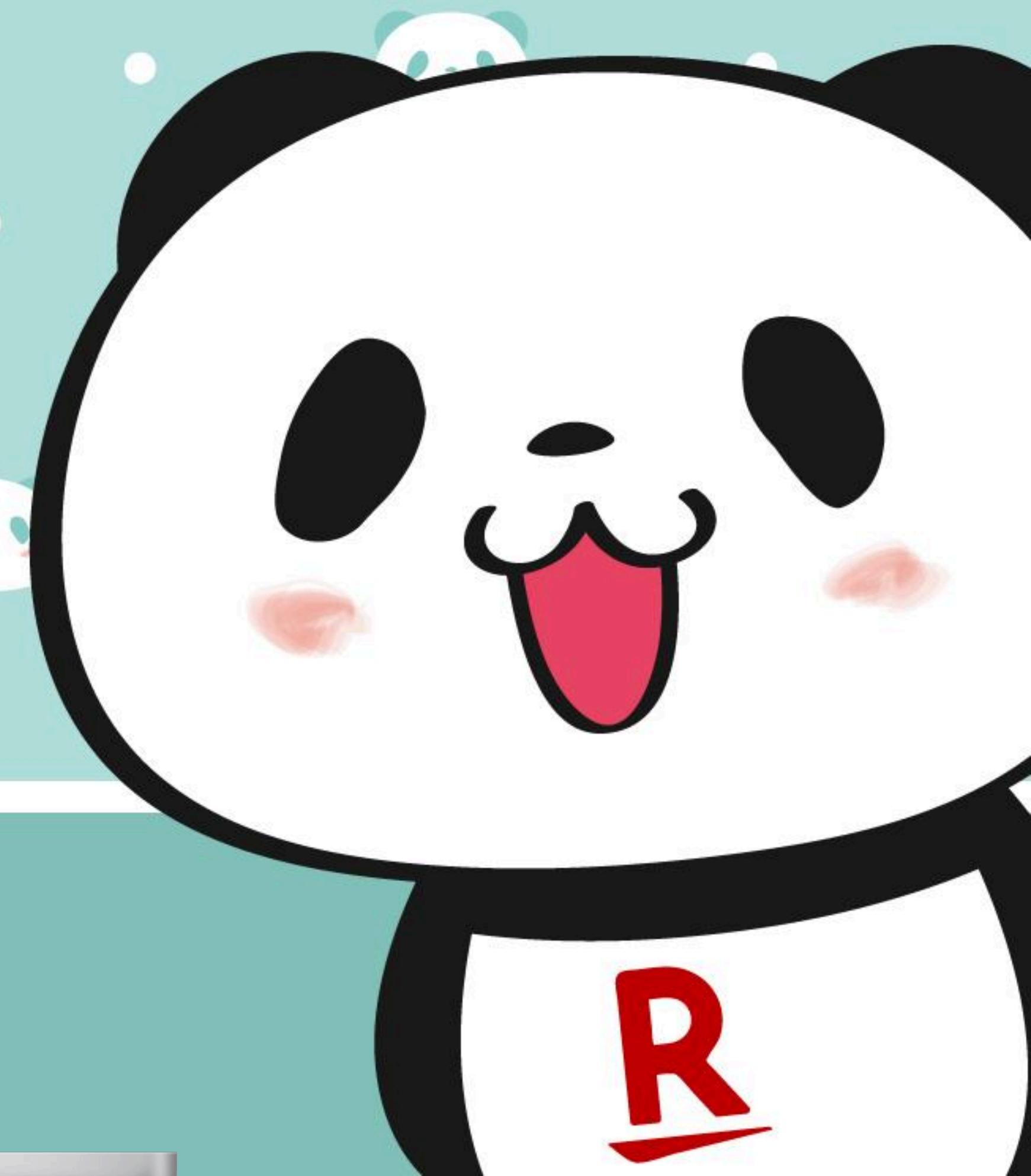
商品番号：0024
当店通常価格 7,590円
7,590円 送料無料
70ポイント(1倍) 内訳

5,590円で購入可! ※送料別
(楽天カード入会&ポイント利用の場合)

商品詳細を選択
梱包・包装：未選択
【ご自宅用】簡易梱包
7,590円

北海道・九州・沖縄エリアは別途送料が発生します。(北海道/ 必須
九州500円、沖縄県1,000円)
了解

数量
- 1 +



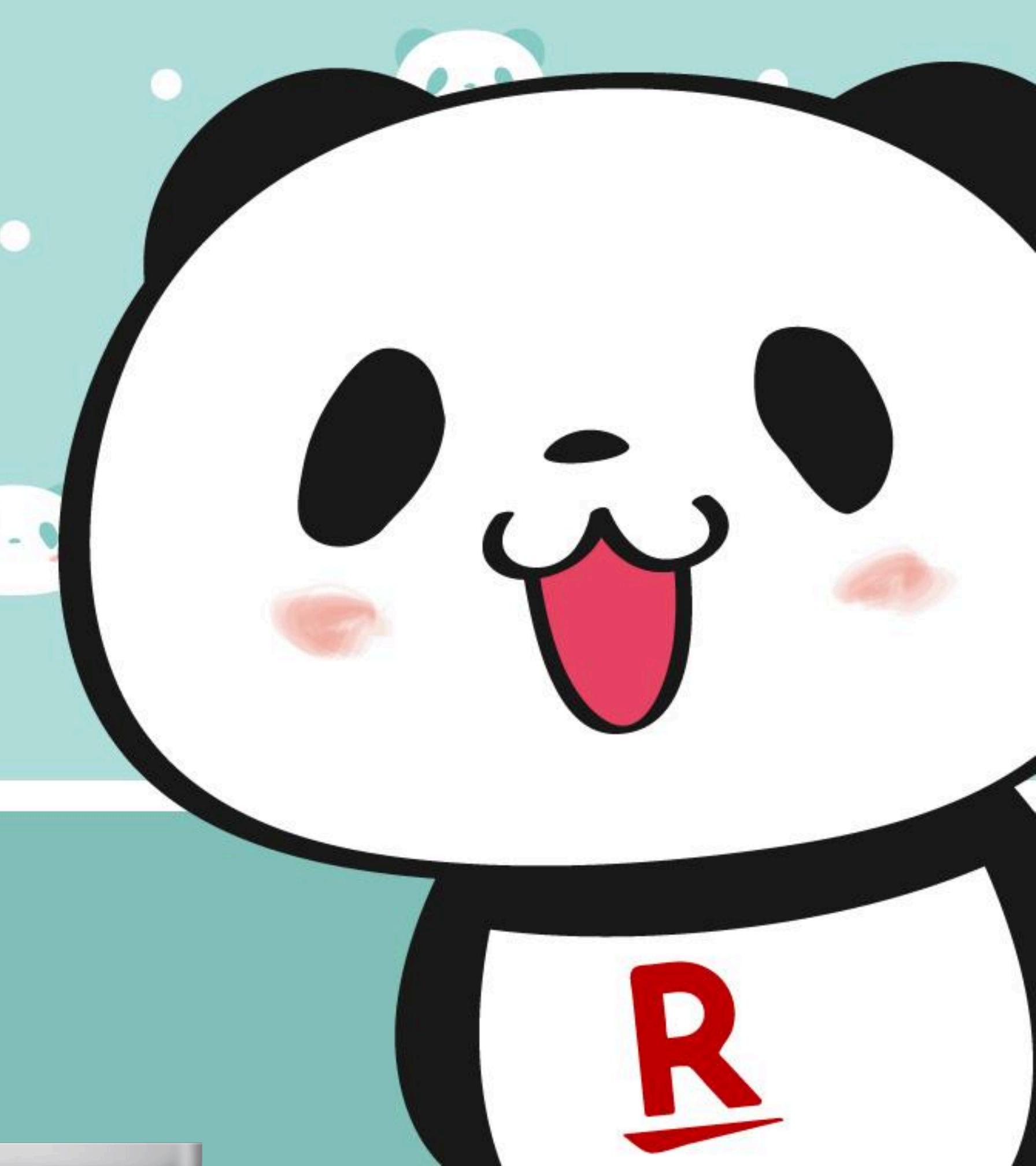
★★★★★ 4.79 (29,094件)  レビューを書く

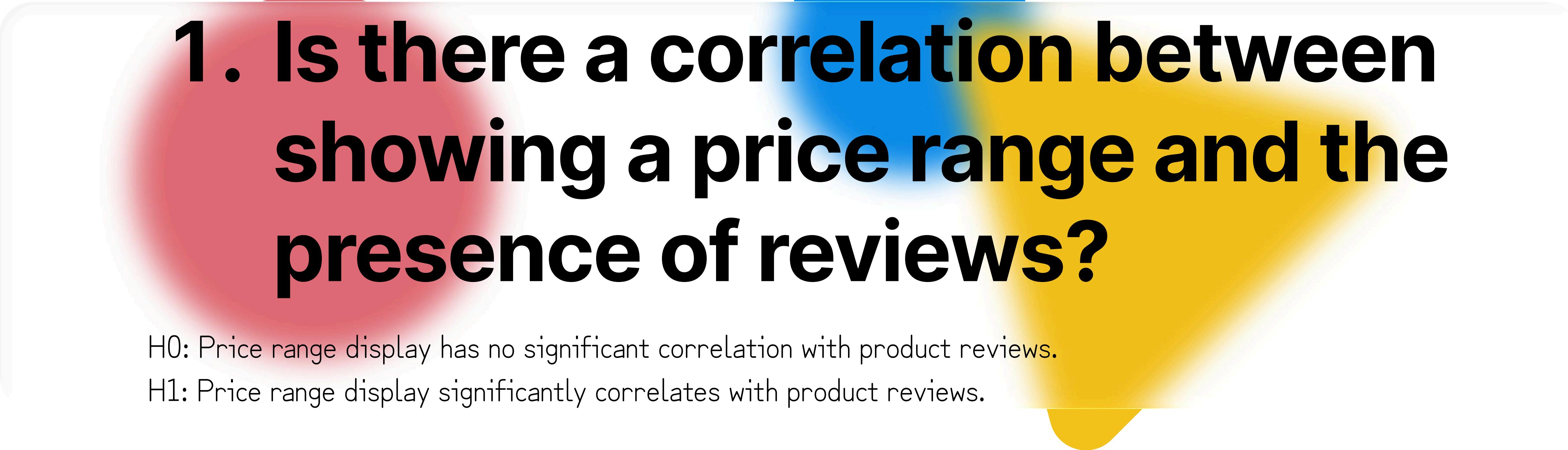
☆ お気に入り商品

☆ お気に入りショップ

シェア  ROOMに投稿

商品についてのお問い合わせ  不適切な商品を報告





1. Is there a correlation between showing a price range and the presence of reviews?

H0: Price range display has no significant correlation with product reviews.

H1: Price range display significantly correlates with product reviews.



```
1 contingency = pd.crosstab(df['has_review'], df['hasPriceRange'])  
2 contingency
```

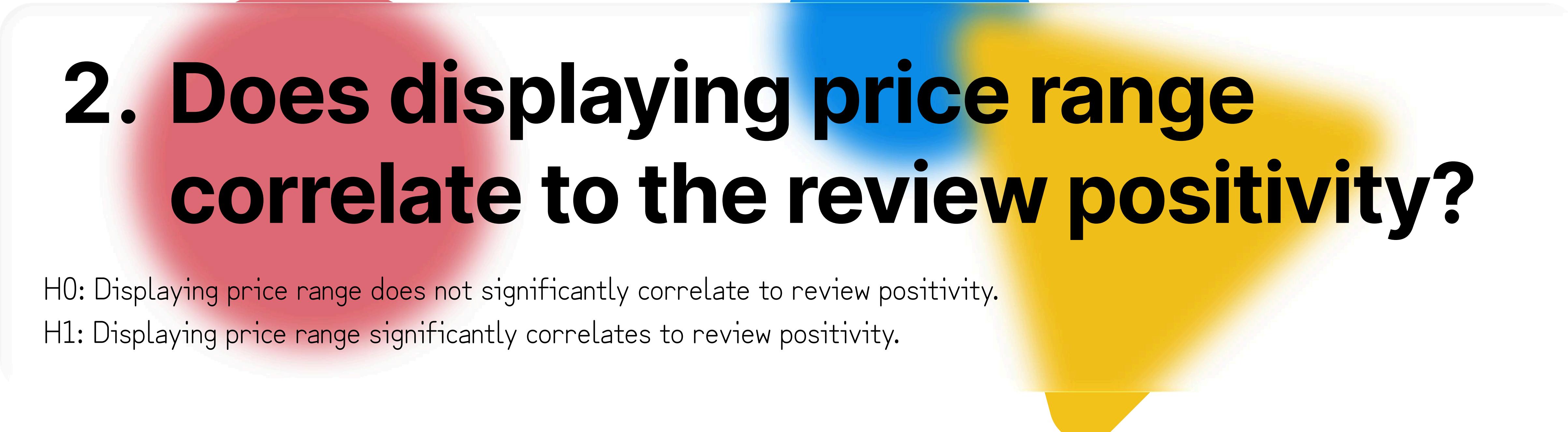
hasPriceRange	0	1
has_review		
0	119	8
1	574	298



```
1 chi2, p, df, expected = stats.chi2_contingency(contingency)  
2 p  
3 >>>3.7538164818492824e-10
```

Dismiss H0:

Indicating a correlation between price range and review presence.



2. Does displaying price range correlate to the review positivity?

H0: Displaying price range does not significantly correlate to review positivity.

H1: Displaying price range significantly correlates to review positivity.



```
1 reviewAverage_on_priceRange = has_review.groupby(  
2     'hasPriceRange')['reviewAverage'].apply(list)
```



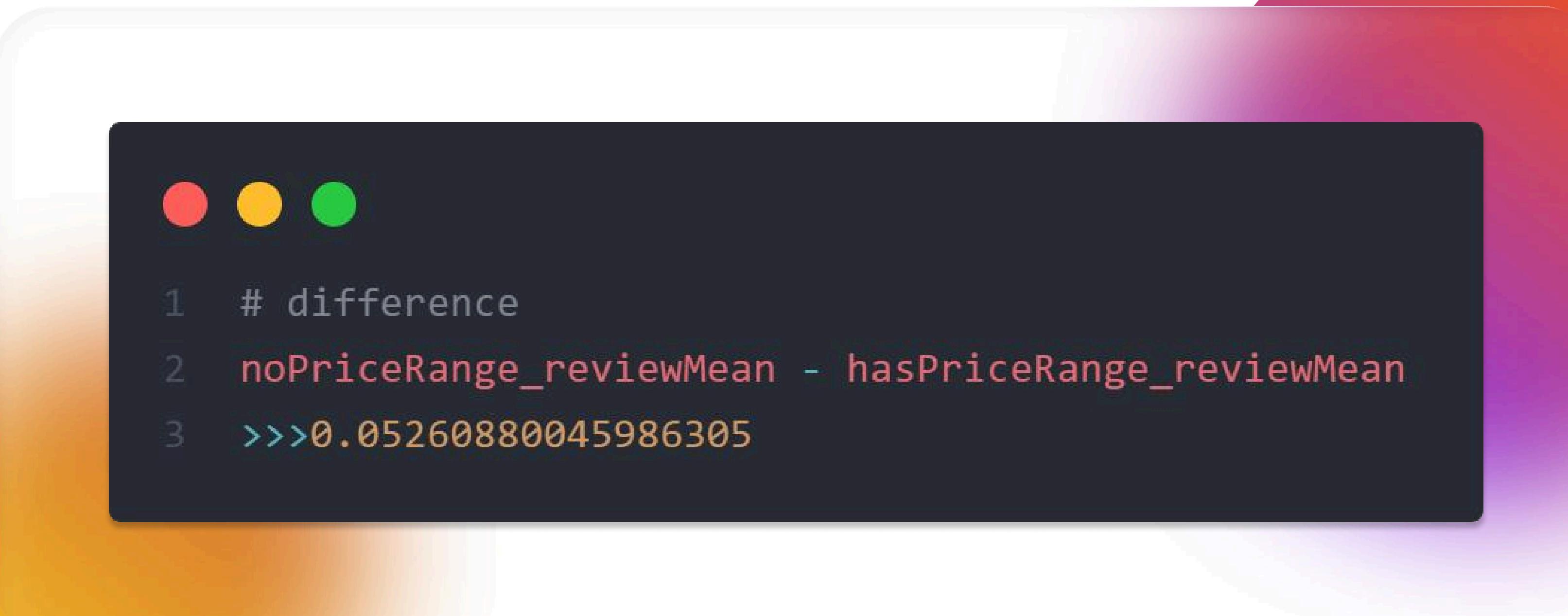
```
1 t_test, p_value = stats.ttest_ind(*reviewAverage_on_priceRange)  
2 p_value  
3 >>>np.float64(0.0019645054959108437)
```



```
1 wow, that was unexpected
```

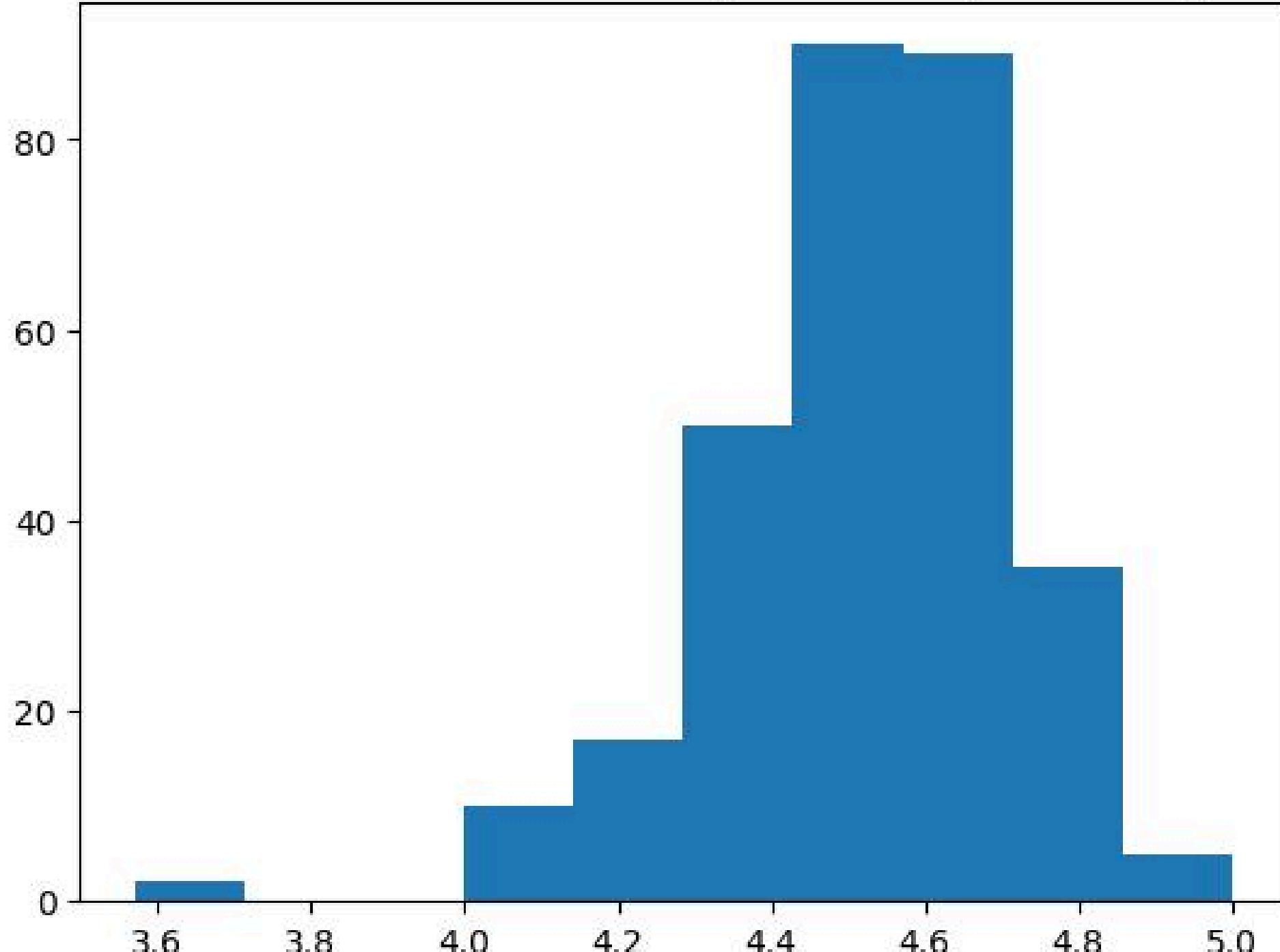
Reject H0:

i.e., there is a correlation between having price range and review positivity.

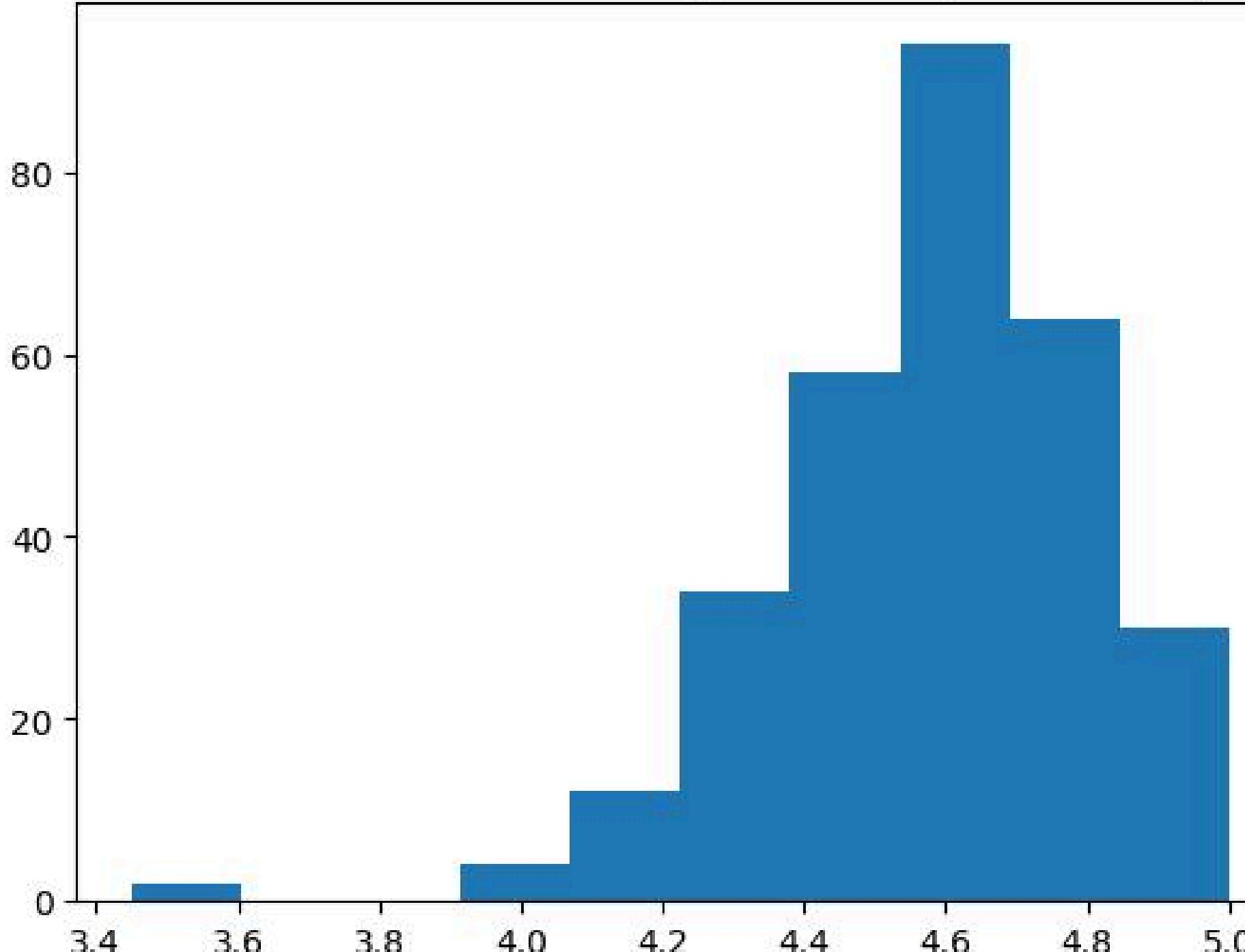


```
● ● ●  
1 # difference  
2 noPriceRange_reviewMean - hasPriceRange_reviewMean  
3 >>>0.05260880045986305
```

Distribution of Review Average of Item w/ Price Range



Distribution of Review Average of Item w/o Price Range

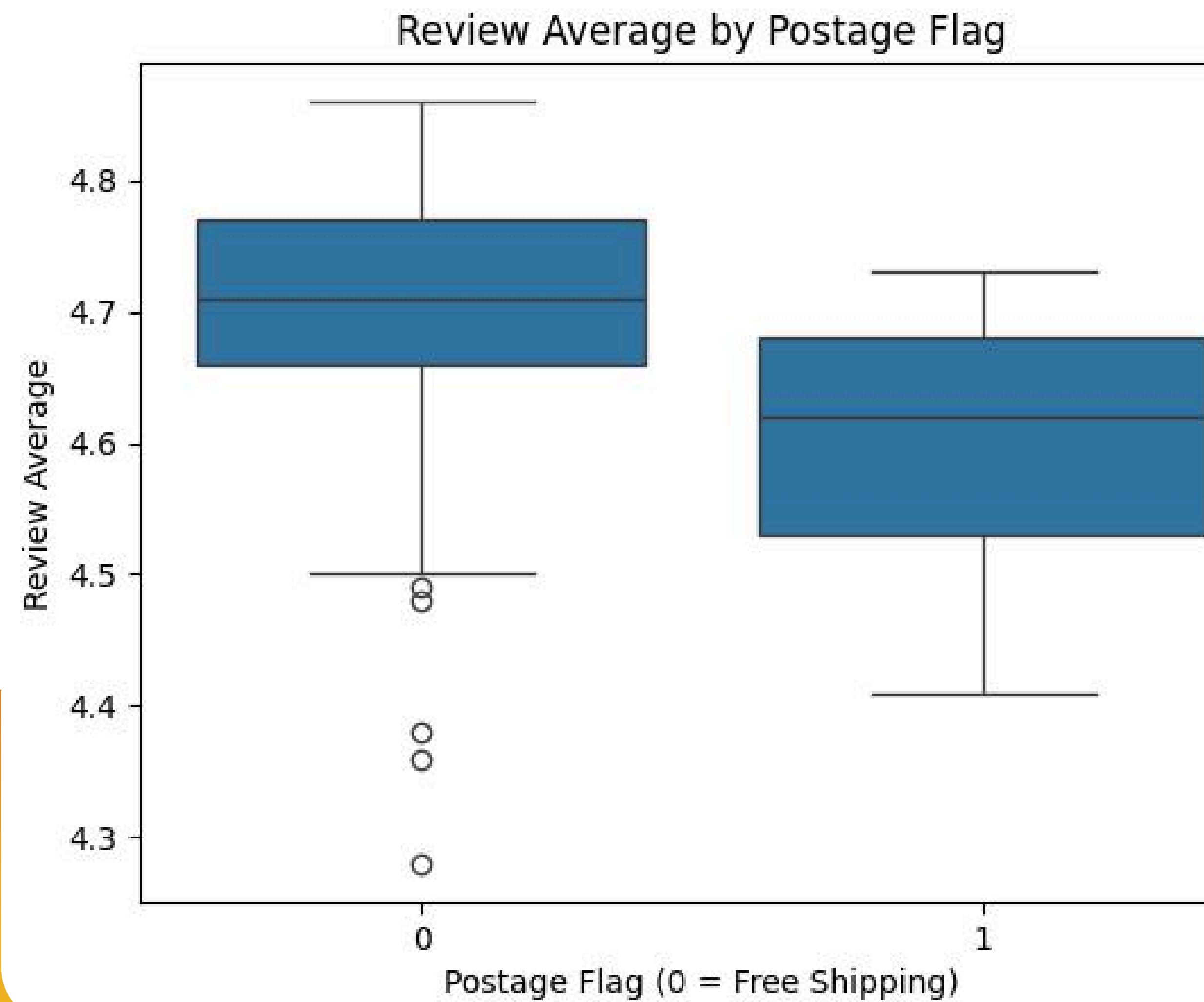


3. Postage Flag and Review Average

- 0: Postage included (送料無料, free shipping)
- 1: Postage not included (buyer pays shipping)

H0: there is no difference in review average between items with and without free shipping.

H1: items with free shipping have higher review averages.



count

postageFlag	count
0	93
1	11

dtype: int64



```
1 free = lens_df[lens_df['postageFlag'] == 0]['reviewAverage']
2 not_free = lens_df[lens_df['postageFlag'] == 1]['reviewAverage']
3
4 t_stat, p_value = ttest_ind(free, not_free)
5
6 print(f't-statistic: {t_stat:.4f}')
7 print(f'p-value: {p_value:.4f}')
```



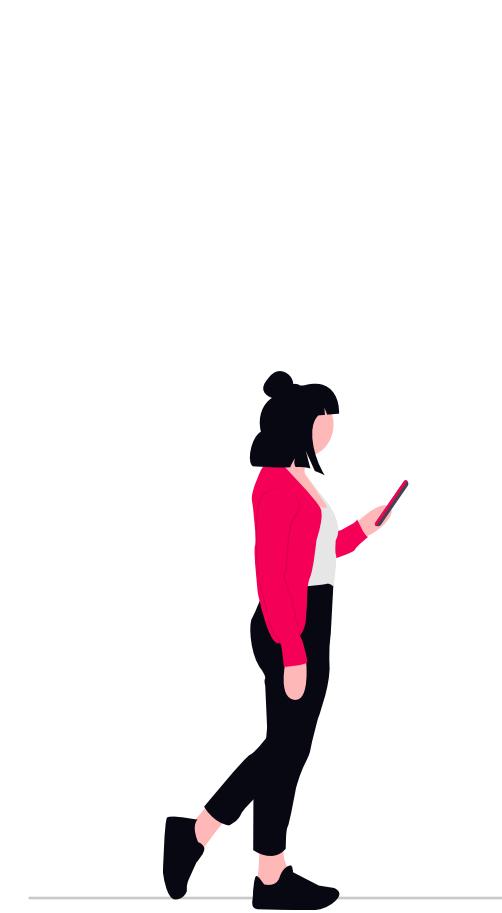
```
1 >>>p-value: 0.0291
```

Reject H0:

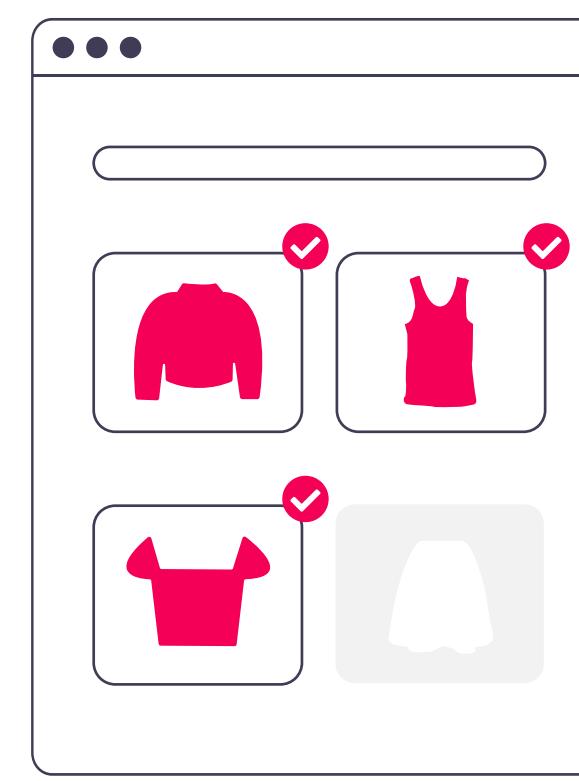
i.e., there is a significant correlation between free shipping and review positivity.

Conclusion

How product display features influence consumer engagement on Rakuten Ichiba.



Displaying Price Range ↔ More Reviews



Displaying Price Range ↔ Better Reviews



Free shipping ↔ Better Reviews



“I got to brush up my knowledge about contingency table and chi²”

Zach



“Got hands-on with an API and tested ideas through real data.”

Jenny



“Approaching data science from python perspective reveals new challenges..”

Sherman

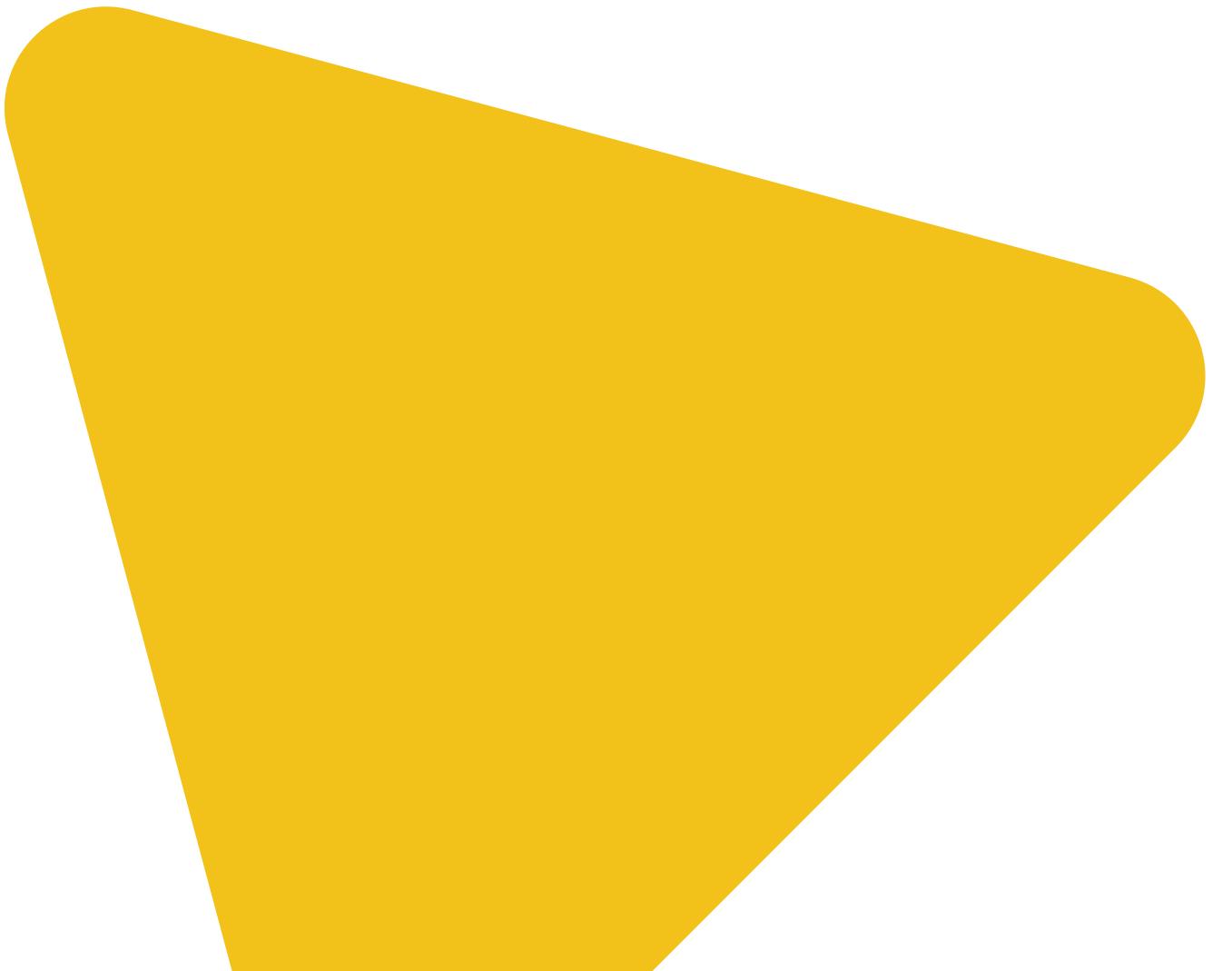
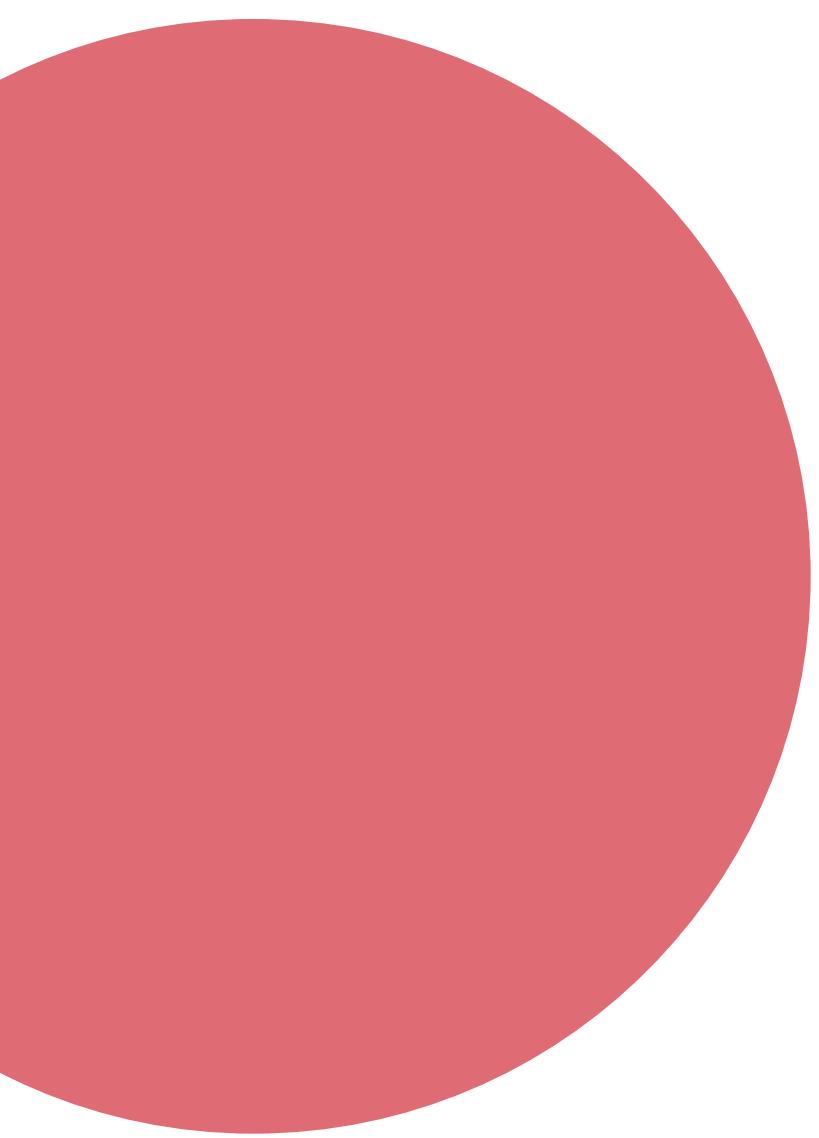
楽天市場

Thank you!



Backup slides





Necessary Libraries

Necessary Libraries

```
import requests  
import json  
import pandas as pd  
from scipy import stats  
import numpy as np  
import matplotlib.pyplot as plt
```

Necessary Libraries

```
params = {  
    'applicationId': APP_ID,  
    'format': 'json',  
    'keyword': '-reviewCount'  
}
```

Rakuten Ichiba Data

楽天市場データ