

ĐẠI HỌC HUẾ
KHOA KỸ THUẬT VÀ CÔNG NGHỆ
๐๐๐๐๐๐



BÁO CÁO ĐỒ ÁN
Học kỳ I, năm học 2021 - 2022
Học phần:
PHÂN TÍCH DỮ LIỆU - R

Số phách

(Do hội đồng chấm thi ghi)

Thừa Thiên Huế, tháng 01 năm 2024

ĐẠI HỌC HUẾ
KHOA KỸ THUẬT VÀ CÔNG NGHỆ
📖📚📖



BÁO CÁO ĐỒ ÁN
Học kỳ I, năm học 2021 - 2022
Học phần:
PHÂN TÍCH DỮ LIỆU - R

Giảng viên hướng dẫn: TS. Hồ Quốc Dũng
Lớp: Khoa học dữ liệu và trí tuệ nhân tạo - K3
Sinh viên thực hiện: Nguyễn Văn Minh Khánh
(ký và ghi rõ họ tên)

Số phách

(Do hội đồng chấm thi ghi)

Thừa Thiên Huế, tháng 01 năm 2024

LỜI CẢM ƠN

“Em đã rất cố gắng và nỗ lực trong bài báo cáo đồ án này. Tuy nhiên, sẽ không thể thực hiện được nếu không có sự hỗ trợ, giúp đỡ ân cần của giảng viên bộ môn – Phân tích dữ liệu R cũng như Ban giám hiệu Khoa Kỹ thuật và Công nghệ - Đại học Huế vì đã tạo điều kiện về cơ sở vật chất, như môi trường học tập thân thiện, giúp em phát huy hết khả năng học tập và rèn luyện nhân cách một cách hiệu quả.

Em muốn bày tỏ lòng biết ơn chân thành đối với giảng viên bộ môn và toàn thể giáo viên của Khoa Kỹ thuật và Công nghệ - Đại học Huế.

Em muốn bày tỏ lòng biết ơn đến gia đình và bạn bè vì đã luôn đồng hành, động viên và quan tâm em trên con đường học tập và trong cuộc sống.

Và lời cảm ơn đặc biệt cuối cùng em xin dành tặng cho bản thân chính mình vì đã không bỏ cuộc vào những lúc bản thân suy sụp, mệt mỏi nhất, cảm ơn bản thân đã luôn cố gắng để vượt qua những khó khăn tưởng chừng không thể bước tiếp, cảm ơn vì tất cả.”

DANH MỤC HÌNH ẢNH

Hình 1: Bảng DevTools	6
Hình 2: Network > Fetch/XHR	6
Hình 3: list request - crawl id product	7
Hình 4: Headers và params - crawl id product	7
Hình 5: API - crawl id product	8
Hình 6: Thông tin sản phẩm cần crawl	9
Hình 7: Tổng quan về tập dữ liệu	13
Hình 8: Kết quả hiển thị thư mục làm việc	14
Hình 9: Số hàng và số cột của tập dữ liệu	14
Hình 10: Trả về tên các trường dữ liệu	15
Hình 11: Dataframe mới	15
Hình 12: Các dòng bị trùng lặp	16
Hình 13: Hiển thị lại DataFrame không chứa dòng trùng lặp	16
Hình 14: Thống kê mô tả tập dữ liệu	17
Hình 15: Biểu đồ cột	18
Hình 16: Biểu đồ tròn	20
Hình 17: Biểu đồ histogram	20
Hình 18: Kết quả kiểm định t	21
Hình 19: Kết quả kiểm tra giả thuyết	23
Hình 20: Tóm tắt mô hình Linear Regression	23
Hình 21: Biểu đồ Linear Regression	25
Hình 22: Thông tin phân cụm dữ liệu	26
Hình 23: Biểu đồ KMeans	27
Hình 24: Kết quả PCA	28
Hình 25: Biểu đồ trực quan PCA	29

DANH MỤC BẢNG BIỂU

MỤC LỤC

LỜI CẢM ƠN.....	i
DANH MỤC HÌNH ẢNH.....	ii
DANH MỤC BẢNG BIỂU	iii
MỤC LỤC	iv
CHƯƠNG 1: MÔ TẢ KỸ THUẬT CRAWL DỮ LIỆU	5
1.1 Kỹ thuật crawl dữ liệu	5
1.1.1 Crawl id sản phẩm từ danh mục thời trang nam	5
Crawl thông tin sản phẩm từ id	9
1.2 Giới thiệu về tập dữ liệu đã crawl	13
CHƯƠNG 2: PHÂN TÍCH TỔNG QUAN VỀ DỮ LIỆU	14
2.1 Đọc dữ liệu	14
2.1.1 Nhập các thư viện cần thiết	14
2.1.2 Thông tin tập dữ liệu	14
2.2 Tiền xử lý dữ liệu	15
2.3 Thống kê mô tả.....	17
2.4 Trực quan hóa dữ liệu.....	18
2.4.1 Biểu đồ cột.....	18
2.4.2 Biểu đồ tròn	19
2.4.2 Biểu đồ histogram.....	20
CHƯƠNG 3: CHỨNG MINH VÀ PHÂN TÍCH CÁC GIẢ THUYẾT, BÀI TOÁN	21
3.1 Chứng minh giả thuyết (Hypothesis Testing)	21
3.2 Dự đoán	23
3.2.1 Linear Regression.....	23
3.2.2 K-Means	25
3.3 Phân tích PCA	28
LINK MÃ NGUỒN.....	30
TÀI LIỆU THAM KHẢO	31
KẾT QUẢ KIỂM TRA ĐẠO VĂN	32

CHƯƠNG 1: MÔ TẢ KỸ THUẬT CRAWL DỮ LIỆU

1.1 Kỹ thuật crawl dữ liệu

API (Application Programming Interface), là 1 giao diện mà giúp chúng ta thu thập và gửi dữ liệu sử dụng code. Chúng ta sử dụng APIs nhiều nhất khi thu thập dữ liệu, và điều đó sẽ được đề cập đến trong bài đồ án này.

Khi chúng ta muốn crawl dữ liệu từ API, chúng ta cần tạo 1 request. Requests được sử dụng ở hầu hết mọi website. Ví dụ, khi bạn truy cập một trang web nào đó, trình duyệt của bạn gửi 1 request đến server của trang web cần crawl. API request hoạt động với cách thức chính xác như vậy: bạn tạo 1 request đến API server để lấy dữ liệu, và server trả về cho bạn thứ bạn muốn.

Ở [chương 1](#) này, chúng ta sẽ crawl dữ liệu từ trang web tiki.vn, chủ yếu để lấy danh sách các sản phẩm (id, tên sản phẩm, giá cả,...) và lưu chúng vào một tệp CSV. Để thực hiện một yêu cầu (request) API, chúng ta sẽ sử dụng ngôn ngữ lập trình là Python để hỗ trợ gửi HTTP requests.

1.1.1 Crawl id sản phẩm từ danh mục thời trang nam

Để lấy được những thông tin đó thì đầu tiên chúng ta phải cần crawl id của từng sản phẩm trước.

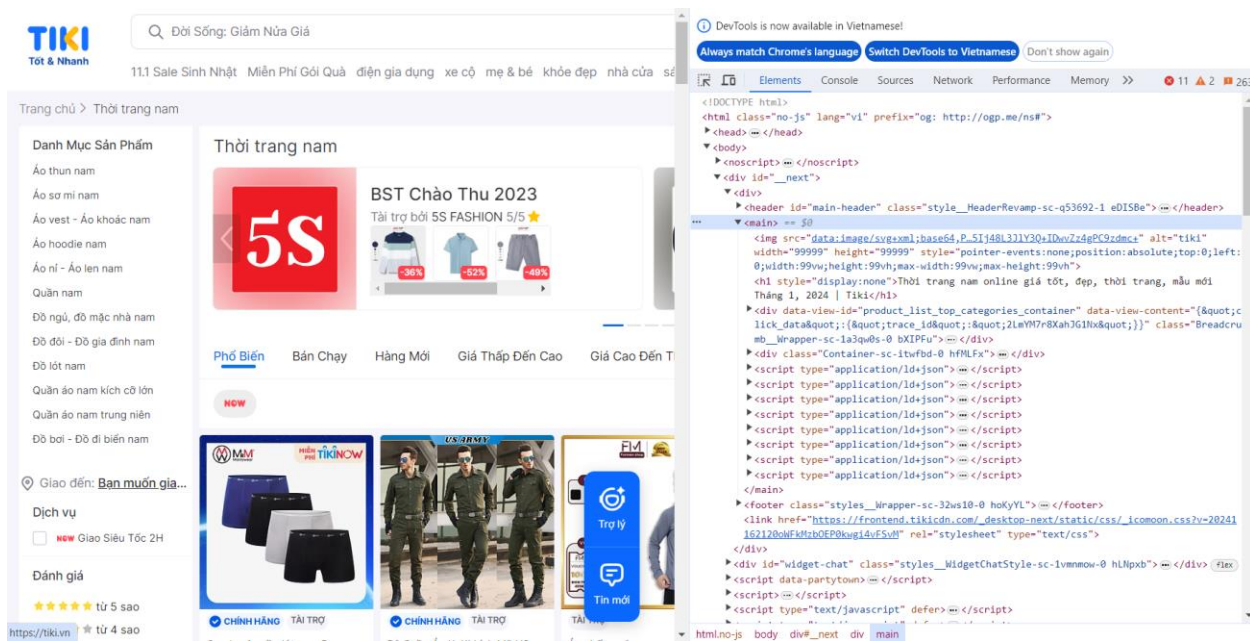
Link trang web crawl id sản phẩm: <https://tiki.vn/thoi-trang-nam/c915>.

Nhập các thư viện cần thiết.

```
import requests # gửi HTTP requests và nhận HTTP responses
import time # tạo độ trễ giữa các requests
import random # tạo ngẫu nhiên thời gian chờ
import pandas as pd # làm việc với dữ liệu dạng bản
```

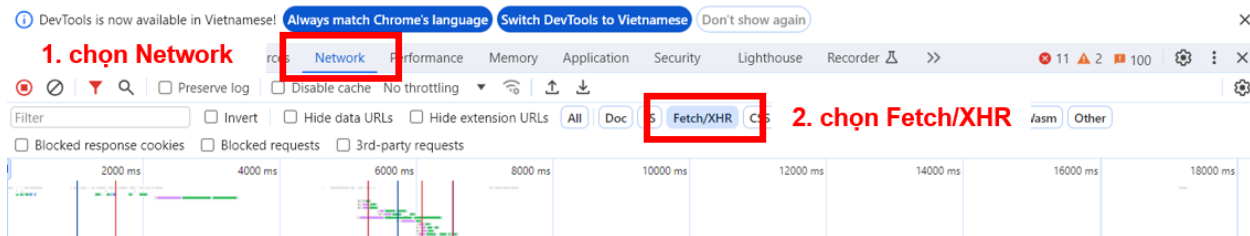
Tiếp đến chúng ta cần nhập header và params của trang web, để lấy được các thông số trong header và params. Ta cần làm theo các bước như sau:

Nhấn chuột phải → Kiểm tra hoặc nhấn phím F12 để mở bảng DevTools.



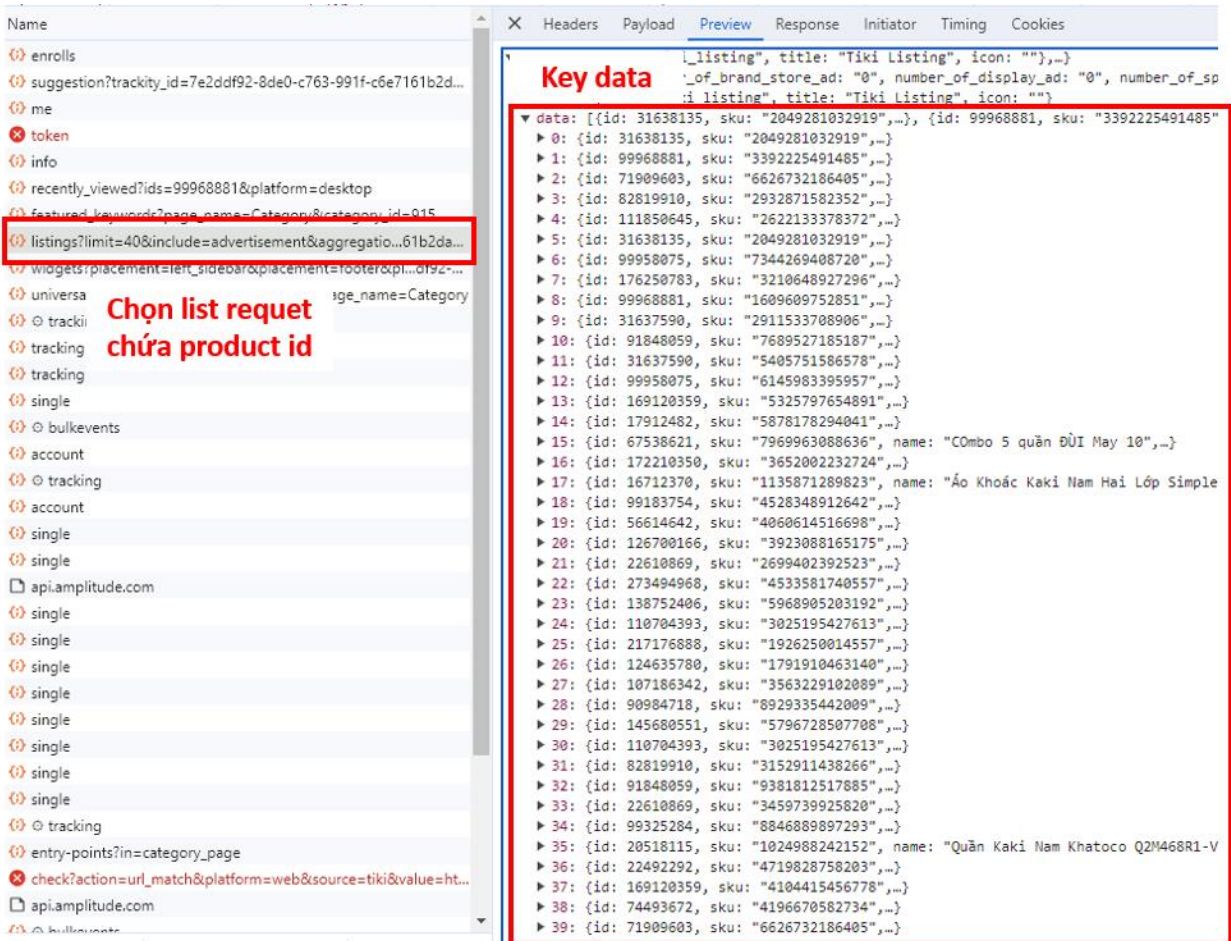
Hình 1: Bảng DevTools

Từ bảng DevTools, chọn Network → tick vào ô Fetch/XHR để hiện các list request (nếu không hiện thì tải lại trang)



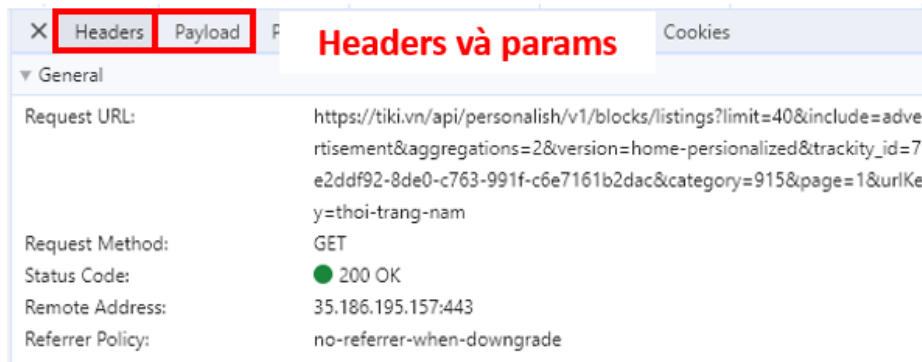
Hình 2: Network > Fetch/XHR

Chọn list request chứa id sản phẩm, nhấn mũi tên bên trái của key data → xuất hiện các list id sản phẩm.



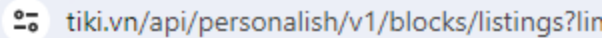
Hình 3: list request - crawl id product

Để crawl được id chúng ta phải lấy thông tin header và params cũng như API của trang web.



Hình 4: Headers và params - crawl id product

Các thông số nằm ở phía dưới. Còn để lấy API thì nhấn đúp vào list request chứa id → xuất hiện cửa sổ mới → copy lại API nằm trên thanh địa chỉ.



Hình 5: API - crawl id product

Bắt đầu crawl dữ liệu, ở đây ta sẽ lấy dữ liệu từ 10 trang.

```
headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36',
    'Accept': 'application/json, text/plain, */*',
    'Accept-Language': 'vi-VN,vi;q=0.9,fr-FR;q=0.8,fr;q=0.7,en-
US;q=0.6,en;q=0.5',
    'Referer': 'https://tiki.vn/thoi-trang-nam/c915',
    'x-guest-token': '8YcN0C1W8zP6ElLJkTgdAMyxrpvQwb94q',
    'Connection': 'keep-alive',
    'TE': 'Trailers',
}

params = {
    'limit': '40',
    'include': 'sale-
attrs,badges,product_links,brand,category,stock_item,advertisement',
    'aggregations': '2',
    'trackity_id': '7e2ddf92-8de0-c763-991f-c6e7161b2dac',
    'category': '915',
    'page': '1',
    'src': 'c915',
    'urlKey': 'thoi-trang-nam',
}

product_id = []
for i in range(1, 11):
    params['page'] = i
    response =
requests.get('https://tiki.vn/api/personalish/v1/blocks/listings',
headers=headers, params=params)#, cookies=cookies)
    if response.status_code == 200:
        print('Request success!!!')
        for record in response.json().get('data'):
            product_id.append({'id': record.get('id')})
        time.sleep(random.randrange(3, 10))
```

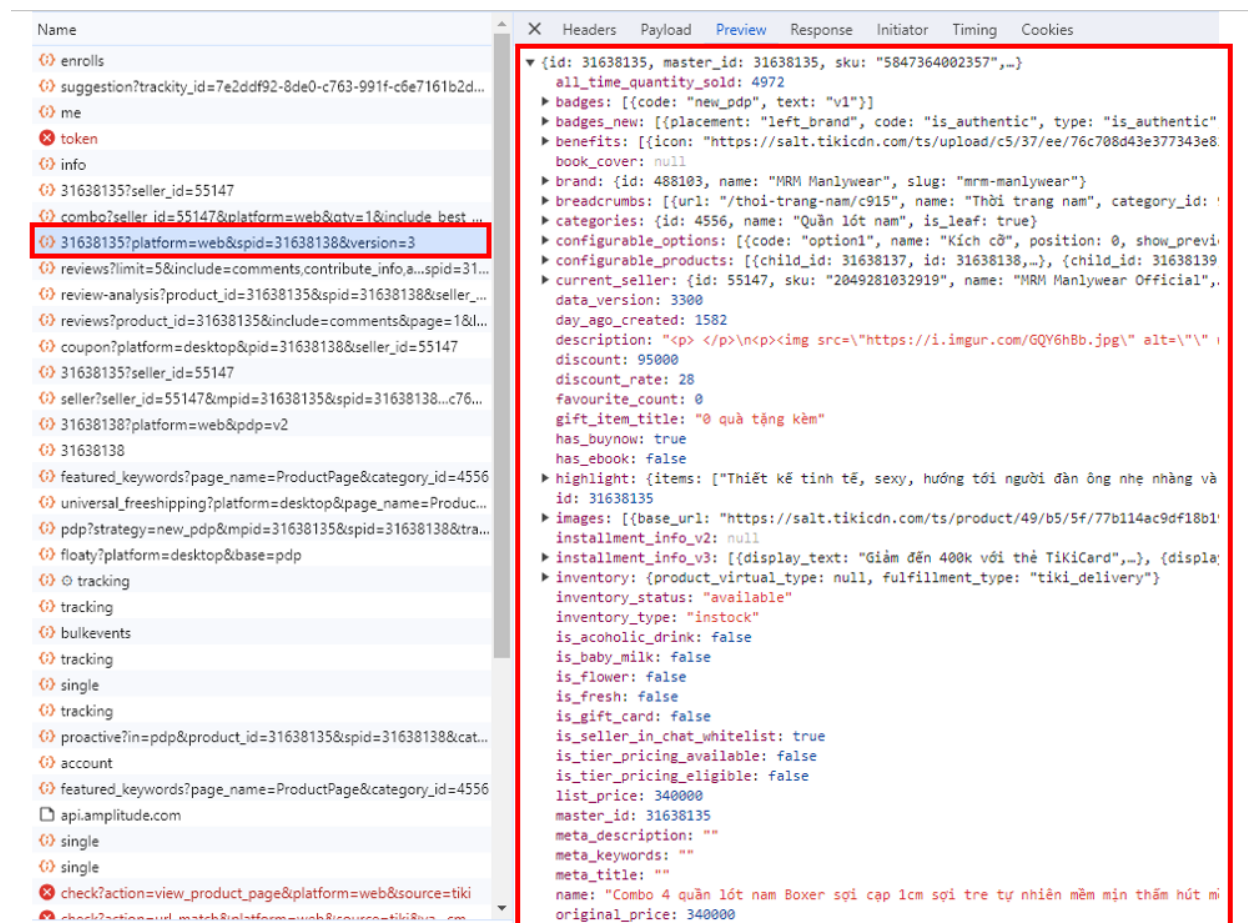
Sau khi crawl thành công thì lưu id sản phẩm vào file csv.

```
df = pd.DataFrame(product_id)
df.to_csv('crawl_product_id.csv', index=False)
```

Crawl thông tin sản phẩm từ id

Để crawl được thông tin sản phẩm thì ta nhân chọn một sản phẩm bất kì để chuyển sang trang mới.

Mở DevTools rồi làm như các bước cũ. Lúc này phải chọn request chứa thông tin sản phẩm như bên dưới. Ở phần Preview sẽ chứa các trường dữ liệu cần crawl (giá cả, tên sản phẩm, đánh giá,...)



Hình 6: Thông tin sản phẩm cần crawl

Phương thức lấy API, headers, params và cookies như bước crawl id.

Điều chỉnh thông số cho cookie, headers và params của trang.

```
import pandas as pd
import requests
```

```

import time
import random
from tqdm import tqdm

cookies = {
    'TIKI_RECOMMENDATION': '6daf0ec0d8aeabb3461783efada3ac72',
    'TKSESSID': 'b971a21329324f186bf7f28716e4b6cf',
    'TOKENS': '{%22access_token%22:%22YcN0C1W8zP6ElLJkTgdAMyxrpvQwb94q%22}',
    '__IP': '1746718024',
    '__R': '0',
    '__iid': '749',
    '__iid': '749',
    '__su': '0',
    '__su': '0',
    '__tb': '0',
    '__uidac': '01659576d001836f069b27e32d7a7499',
    '__uif': '__uid%3A7642940961746729271%7C__ui%3A-1%7C__create%3A1704294096',
    '__utm': 'source%3Dtiki-aff%7Cmedium%3Dtiki-
aff%7Ccampaign%3DAFF_NBR_TIKIAFF_UNK_TIKIVN-
D3K72LS2_ALL_VN_ALL_UNK_UNK_TAPX.f1bf430a-65cf-418a-b43e-
28dfa9168045_TAPU.e8811fd7-587b-4b1d-b685-d9586955086f',
    '__utm': 'source%3Dtiki-aff%7Cmedium%3Dtiki-
aff%7Ccampaign%3DAFF_NBR_TIKIAFF_UNK_TIKIVN-
D3K72LS2_ALL_VN_ALL_UNK_UNK_TAPX.f1bf430a-65cf-418a-b43e-
28dfa9168045_TAPU.e8811fd7-587b-4b1d-b685-d9586955086f',
    '_fbp': 'fb.1.1704294114476.1292406438',
    '_ga': 'GA1.1.1519596821.1704294091',
    '_ga_7QN4MZMLVG': 'GS1.1.1704336969.1.1.1704336971.0.0.0',
    '_ga_L9MPNN6QJB': 'GS1.1.1704336969.1.1.1704336980.0.0.0',
    '_ga_S9GLR1RQFJ': 'GS1.1.1704334874.3.1.1704340301.60.0.0',
    '_ga_W6PZ1YEX5L': 'GS1.1.1704338649.1.1.1704339309.0.0.0',
    '_gcl_a': '1.1.737102976.1704294095',
    '_hjAbsoluteSessionInProgress': '1',
    '_hjIncludedInSessionSample_522327': '0',
    '_hjSessionUser_522327':
'eyJpZCI6IjBkZmM0NDMwLTNmYjctNTViMy1iYzcc0LTc5ZjU1NjQ2Y2QxYyIsImNyZWZlZmQ3MDQyOTQwOTU5NDUsImV4aXN0aw5nIjp0cnVlfQ==',
    '_hjSession_522327':
'eyJpZCI6ImEwNGM2ODJiLWZmZDEtNDU0Ni1iMzJlLTU3NTZiNWJhZjM4ZCIsImMiOjE3MDQzMzMwMTU4MzEsInMiOjAsInIiOjAsInNiIjowfQ',
    '_trackity': '7e2ddf92-8de0-c763-991f-c6e7161b2dac',
    'amp_99d374': '6gD267ETVsbYkVTUmYnfSE...1hj92jli6.1hj99cat5.5s.8i.ee',
    'cto_bundle': 'LnIg-
F83MX1qbUVEb01wcnVNeG5TTG41VkZLUG5vMjdobH1TM1JmTyUyQkt5aXczYTdXaXRta29ROWpVMH1z

```

```

VWZtWnFuRGF5M1BNU0NMCSuyQmdtYlh5QVdTt3gzVk1pUklwSmZaN0h0N1BjZnR2ZURLWDJtazQxWjV
US0hxY1p2S2lJZH1FNkxHQjU1MkY3ZUclMkZTWd1iazFoekc5SWdQQUzRCUzRA',
    'delivery_zone': 'Vk4wMzkwMDYwMDE=',
    'dtdz': '-1',
    'rl_anonymous_id':
'StackityEncrypt%3AU2FsdGVkX19IHHSuW%2Fa7YdscvoeaIiNu1AQ6XK0r6ed3lALdJ00T1fTAs8
oSKVQWGif7Z0QECLkL3%2FZg56mQYw%3D%3D',
    'rl_group_id':
'StackityEncrypt%3AU2FsdGVkX1%2Bzr5RcL6%2BpdRK%2FM6MoTfF%2BqGSXpaziCc4%3D',
    'rl_group_trait ':
'StackityEncrypt%3AU2FsdGVkX19Raa%2FMmvCJG3SIznh%2F1j6yT4M3qjl0dPs%3D',
    'rl_page_init_referrer':
'StackityEncrypt%3AU2FsdGVkX1%2FU4RPIsBYghwLD8N1pAFHzbt6GJ3rv%2BP%2FM%2BBc3tcdL
173VtWCXdyMo',
    'rl_page_init_referring_domain':
'StackityEncrypt%3AU2FsdGVkX19L1VhPZty3IE7%2FHZN5dsBc%2BTaRpnHsj%2F3JfwNfMa%2BV
gsAeIbkgEjxW',
    'rl_trait':
'StackityEncrypt%3AU2FsdGVkX1%2FMH1NKb%2Bqy2ITv0sekVVS0RYOAFe9Xr2c%3D',
    'rl_user_id ':
'StackityEncrypt%3AU2FsdGVkX1%2Bivmh9WZr8a0rYPL0hPrKk%2BqDRivoA7uU%3D',
    'tiki_client_id': '1519596821.1704294091'
}

headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36',
    'Accept': 'application/json, text/plain, */*',
    'Accept-Language': 'vi-VN,vi;q=0.9,fr-FR;q=0.8,fr;q=0.7,en-
US;q=0.6,en;q=0.5',
    'Referer': 'https://tiki.vn/set-2-qua-n-du-i-nam-qua-n-short-gio-nam-the-
thao-basic-tre-trung-nang-do-ng-thoa-ng-ma-t-co-gia-n-4-chie-u-mrm-manlywear-
p99968881.html?itm_campaign=CTP_YPD_TKA_PLA_UNK_ALL_UNK_UNK_UNK_X.284641_Y.
1866961_Z.3907942_CN.03%2F2024---CB-2-Q%C4%90-Tron---
Auto&itm_medium=CPC&itm_source=tiki-ads&spid=99968992',
    'x-guest-token': 'YcN0C1W8zP6ElLJkTgdAMyxrvpQwb94q',
    'Connection': 'keep-alive',
    'TE': 'Trailers',
}

params = (
    ('platform', 'web'),
    ('spid', 99968992)

```

```

    #('include',
    'tag,images,gallery,promotions,badges,stock_item,variants,product_links,discount_tag,ranks,breadcrumbs,top_features,cta_desktop'),
    )

```

Tạo hàm lấy thông tin ở phần Preview.

```

def parser_product(json):
    d = dict()
    d['id'] = json.get('id')
    d['brand_id'] = json.get('brand').get('id')
    d['brand_name'] = json.get('brand').get('name')
    d['product_name'] = json.get('name')
    d['categories'] = json.get('categories').get('name')
    d['discount'] = json.get('discount')
    d['discount_rate'] = json.get('discount_rate')
    d['list_price'] = json.get('list_price')
    d['price'] = json.get('price')
    d['rating_average'] = json.get('rating_average')
    d['review_count'] = json.get('review_count')
    d['stock_item_qty'] = json.get('stock_item').get('qty')
    d['stock_item_max_sale_qty'] = json.get('stock_item').get('max_sale_qty')
    d['short_url'] = json.get('short_url')
    return d

```

Lấy danh sách id sản phẩm để chuẩn bị cho việc crawl.

```

df_id = pd.read_csv('crawl_product_id.csv')
p_ids = df_id.id.to_list()
print(p_ids)
result = []

```

Bắt đầu crawl dữ liệu.

```

for pid in tqdm(p_ids, total=len(p_ids)):
    response = requests.get('https://tiki.vn/api/v2/products/{}'.format(pid),
    headers=headers, params=params, cookies=cookies)
    if response.status_code == 200:
        print('\nCrawl data {} success !!!'.format(pid))
        result.append(parser_product(response.json()))
    # time.sleep(random.randrange(3, 5))
df_product = pd.DataFrame(result)

```

Lưu thông tin sản phẩm vào file csv.

```

df_product.to_csv('crawl_product_data.csv', index=False)

```


1.2 Giới thiệu về tập dữ liệu đã crawl

Đây là tập dữ liệu về các sản phẩm thuộc danh mục thời trang nam trên trang web thương mại điện tử Tiki.vn. Dưới đây là chi tiết các trường dữ liệu đã crawl được.

- **id:** id sản phẩm.
- **brand_id:** id của nhãn hàng.
- **brand_name:** tên nhãn hàng.
- **product_name:** tên sản phẩm.
- **categories:** danh mục.
- **discount:** giảm giá.
- **discount_rate:** tỉ lệ giảm giá.
- **list_price:** giá gốc của sản phẩm.
- **price:** giá sản phẩm sau khi giảm giá.
- **rating_average:** đánh giá trung bình.
- **review_count:** số lượng đánh giá.
- **stock_item_qty:** số lượng hàng tồn kho.
- **stock_item_max_sale_qty:** giá trị lớn nhất của số lượng sản phẩm đã bán (sale quantity) cho mỗi mặt hàng hoặc sản phẩm trong một khoảng thời gian cụ thể.
- **short_url:** link sản phẩm.

Ảnh tổng quan về tập dữ liệu.

id	brand_id	brand_name	product_name	categories	discount	discount_rate	list_price	price	rating_average	review_count	stock_item_qty	stock_item_max_sale_qty	short_url
4027231	111481	DEM	Áo Khoác Dù Nam Phối Dây Kéo - K2015	Thời Trang	0	0	124000	124000	3.9	114	1000	1000	https://tiki.vn/product-p4027231.htm?spid=47686396
7170169	111481	DEM	Bộ 5 Quần Síp Tam Giác Nam Nhật Bản (Nhều miku) - 55Star	Thời Trang	0	0	95000	95000	4.2	487	1000	1000	https://tiki.vn/product-p7170169.htm?spid=7170173
7175397	111481	DEM	Bộ 4 Quần Síp Tam Giác Nam Nhật Bản (Nhều miku) - 55Star	Thời Trang	0	0	85000	85000	3.7	12	1000	1000	https://tiki.vn/product-p7175397.htm?spid=7175417
9730171	219929	CITYMEN	Combo 10 Quần Lót Nam nhiều Lưng vải cotton 2 chiều Hè...	Thời Trang	81000	45	180000	99000	4.5	583	1000	1000	https://tiki.vn/product-p9730171.htm?spid=9730173
11203758	111481	DEM	Combo 4 Quần Síp Đùi Boxer: Thông Hơi Thoáng Mát - Quik...	Thời Trang	0	0	85000	85000	4.5	882	1000	1000	https://tiki.vn/product-p11203758.htm?spid=11203760
14695097	150201	Việt Tiến	Áo sơ mi trắng dài tay Việt Tiến 500	Root	0	0	590000	590000	4.8	14	1000	1000	https://tiki.vn/product-p14695097.htm?spid=47028016
14847948	273717	GOKING	Áo thun nam thoát nhiệt Nhật Bản GOKING siêu thoáng má...	Thời Trang	0	0	159000	159000	4.6	275	1000	1000	https://tiki.vn/product-p14847948.htm?spid=14847950
15784024	317963	NIE	Áo Khoác Nam Áo Khoác Dù Dây Nam Trơn Cao Cấp Shop...	Thời Trang	86000	35	245000	159000	4.5	530	1000	1000	https://tiki.vn/product-p15784024.htm?spid=15784027
15992824	317963	NIE	Áo Khoác Nam Áo Khoác Dù Nam 2 Lấp Cao Cấp-4027	Thời Trang	111000	41	270000	159000	4.5	246	1000	1000	https://tiki.vn/product-p15992824.htm?spid=15992826
16712370	317963	NIE	Áo Khoác Kaki Nam Hài Lấp Simple Cao Cấp ShopH6-KK35	Thời Trang	76000	31	245000	169000	3.9	1151	1000	1000	https://tiki.vn/product-p16712370.htm?spid=16712385
17912482	224521	Doka	Áo khoác nam chống nắng gió thu đông Doka (DBLS1502) c...	Thời Trang	90000	39	229000	139000	4.5	404	1000	1000	https://tiki.vn/product-p17912482.htm?spid=48541088
17912482	224521	Doka	Áo khoác nam chống nắng gió thu đông Doka (DBLS1502) c...	Thời Trang	90000	39	229000	139000	4.5	404	1000	1000	https://tiki.vn/product-p17912482.htm?spid=48541088
17912482	224521	Doka	Áo khoác nam chống nắng gió thu đông Doka (DBLS1502) c...	Thời Trang	90000	39	229000	139000	4.5	404	1000	1000	https://tiki.vn/product-p17912482.htm?spid=48541088
20124380	248413	Heint Boutique	Quần nam ống rộng lưng thun bên bên chất liệu dũi cao cấp...	Thời Trang	0	0	149000	149000	2.5	4	1000	1000	https://tiki.vn/product-p20124380.htm?spid=59176599
20518115	418935	KHATOCO	Quần kaki Nam Khatoco Q2M46861-VNMA012-18115-1 - Ka...	Root	0	0	466000	466000	4.8	160	1000	1000	https://tiki.vn/product-p20518115.htm?spid=20516137
20518655	418935	KHATOCO	Quần Short Nam Khatoco Q2M45180-VNMA031-2009-018 -	Root	0	0	318000	318000	4.8	38	1000	1000	https://tiki.vn/product-p20518655.htm?spid=20516589
20518657	418935	KHATOCO	Quần kaki Nam Khatoco Q2M52850-VNMA020-2006-0_Na...	Root	0	0	528000	528000	5.0	6	1000	1000	https://tiki.vn/product-p20518657.htm?spid=20516870
20521038	418935	KHATOCO	Áo Sơ Mi Nam Tay Ngắn Khatoco A1M4N3871-CHNCR115-2...	Root	0	0	536000	536000	5.0	68	1000	1000	https://tiki.vn/product-p20521038.htm?spid=20521271
20521278	418935	KHATOCO	Áo Sơ Mi Nam Tay Ngắn Khatoco A1M4N3871-CHNCR117-2...	Root	0	0	536000	536000	5.0	43	1000	1000	https://tiki.vn/product-p20521278.htm?spid=20521283
20522765	418935	KHATOCO	Áo Sơ Mi Nam Tay Ngắn Khatoco A1M4N3872-CHN0201-02...	Root	0	0	568000	568000	5.0	2	1000	1000	https://tiki.vn/product-p20522765.htm?spid=20522773
20525612	418935	KHATOCO	Áo Sơ Mi Nam Tay Dài Khatoco A1M4N36851-CHN066-2009...	Root	0	0	568000	568000	5.0	2	1000	1000	https://tiki.vn/product-p20525612.htm?spid=20525615
20525991	418935	KHATOCO	Áo Sơ Mi Nam Tay Dài Khatoco A1M4N36871-CHN0211-200...	Root	0	0	598000	598000	4.8	15	1000	1000	https://tiki.vn/product-p20525991.htm?spid=20525949
20574575	418935	KHATOCO	Áo Sơ Mi Nam Tay Dài Khatoco A1M4N44851-CHN071-2009...	Root	0	0	468000	468000	0.0	0	1000	1000	https://tiki.vn/product-p20574575.htm?spid=20574579
20991311	150201	Việt Tiến	Áo sơ mi trơn tay ngắn	Thời Trang	0	0	300000	300000	4.0	12	1000	1000	https://tiki.vn/product-p20991311.htm?spid=20991359
21601483	111481	DEM	Combo 4 quần lót nam quần sịp tam giác nam đức không đ...	Thời Trang	0	0	319000	319000	4.3	69	1000	1000	https://tiki.vn/product-p21601483.htm?spid=21601485
21659759	111481	DEM	Áo Khoác da nam lốt lông thú đông phối 2 túi AN63	Root	0	0	298000	298000	5.0	1	1000	1000	https://tiki.vn/product-p21659759.htm?spid=46206915
22492292	248545	doisafashion	Áo polo nam ngắn tay cổ cổ (Tăng 1 quần lót nam) , Comb...	Thời trang nam	126000	44	285000	159000	4.5	490	1000	1000	https://tiki.vn/product-p22492292.htm?spid=22492294
22610869	234521	Doka	Combo 3 Áo Thun nam HÀNG HIỆU da phong cách - 683Q...	Thời trang nam	111000	37	300000	189000	4.5	216	1000	1000	https://tiki.vn/product-p22610869.htm?spid=179966391
22854030	205249	Nanjiren	Hộp 4 quần lót nam Boxer Nanjiren quần sịp đùi 1 hàng nh...	Root	163000	41	400000	235000	4.7	428	1000	1000	https://tiki.vn/product-p22854030.htm?spid=22854036
22760941	248545	doisafashion	Combo 3 Áo thun nam cổ bẻ (Tăng 1 áo thun cổ tròn hàng...	Root	127000	31	406000	279000	4.5	222	1000	1000	https://tiki.vn/product-p22760941.htm?spid=72322287
23014297	111481	DEM	Áo Sơ Mi Nam Dài Tay Công Sô, Chất Liệu Chống Nấm Cao...	Thời Trang	0	0	97000	97000	4.7	3	1000	1000	https://tiki.vn/product-p23014297.htm?spid=740305519
23367632	111481	DEM	Quần bơi nam phồng thùng	Thời Trang	0	0	80000	80000	4.9	12	1000	1000	https://tiki.vn/product-p23367632.htm?spid=74037088
23403301	111481	DEM	Quần đùi đồ nam	Thời Trang	0	0	128000	128000	4.9	13	1000	1000	https://tiki.vn/product-p23403301.htm?spid=110472336
23464722	111481	DEM	Quần jogger thun trơn nam (Ben)	Thời Trang	0	0	78000	78000	4.5	2	1000	1000	https://tiki.vn/product-p23464722.htm?spid=77966217

Hình 7: Tổng quan về tập dữ liệu

CHƯƠNG 2: PHÂN TÍCH TỔNG QUAN VỀ DỮ LIỆU

2.1 Đọc dữ liệu

2.1.1 Nhập các thư viện cần thiết

Nhập các thư viện cần thiết.

```
library(dplyr)
library(ggplot2)
```

2.1.2 Thông tin tập dữ liệu

Hiển thị thư mục làm việc.

```
getwd()
```

Kết quả:

```
[1] "C:/Users/Karrot/Documents/Đại học/Năm 2/Crawl Data - R/tiki_crawldata"
```

Hình 8: Kết quả hiển thị thư mục làm việc

Nhập tập dữ liệu.

```
# Nhập tập dữ liệu
df <- read.csv("C:/Users/Karrot/Documents/Đại học/Năm 2/Crawl Data -
R/tiki_crawldata/crawl_product_data.csv")
```

In ra số hàng và số cột của tập dữ liệu.

```
# Trả về số hàng và số cột
dim(df)
```

Kết quả:

```
[1] 430 14
```

Hình 9: Số hàng và số cột của tập dữ liệu

In ra tên của các trường dữ liệu.

```
# Trả về tên các cột
names(df)
```


Kết quả:

```
[1] "id"          "brand_id"      "brand_name"
[9] "price"       "rating_average" "review_count"
      "product_name" "categories"    "discount"
      "stock_item_qty" "stock_item_max_sale_qty" "short_url"
      "discount_rate"  "list_price"
```

Hình 10: Trả về tên các trường dữ liệu

2.2 Tiền xử lý dữ liệu

Xóa các cột không cần thiết.

```
df <- df %>% select
(-brand_id, -product_name, -stock_item_qty, -stock_item_max_sale_qty, -short_url)

# In kết quả
print(df)
```

Kết quả:

```
[1] "id"          "brand_name"    "categories"    "discount"      "discount_rate"
      "list_price"  "price"         "rating_average" "review_count"
```

Hình 11: Dataframe mới

Tìm và xóa các giá trị bị trùng lặp.

```
# Tìm các dòng bị trùng lặp trong DataFrame
row_duplicated <- df[duplicated(df), ]

# Hiển thị DataFrame chứa các dòng bị trùng lặp
print(row_duplicated)
```

Kết quả:

	id <int>	brand_name <chr>	categories <chr>
6	31638135	MRM Manlywear	Quần lót nam
11	99968881	MRM Manlywear	Quần thun nam ngắn
15	99958075	MRM Manlywear	Quần thun nam ngắn
32	82819910	ARADO FASHION	Áo thun nam ngắn tay có cổ
43	126700166	LADOS	Quần jeans nam dài
56	126701222	LADOS	Quần tây nam
57	17912482	Doka	Thời Trang
58	111850645	US ARMY	Đồ mặc nhà nam - Bộ dài
59	205162616	CITYMEN	Quần lót nam
60	99958075	MRM Manlywear	Quần thun nam ngắn

Hình 12: Các dòng bị trùng lặp

Loại bỏ giá trị trùng lặp.

```
# Loại bỏ các dòng trùng lặp, giữ lại dòng đầu tiên xuất hiện
df <- unique(df)

# Hiển thị lại DataFrame không chứa dòng trùng lặp
print(df)
```

Kết quả:

	id <int>	brand_name <chr>
1	31638135	MRM Manlywear
2	82819910	ARADO FASHION
3	111850645	US ARMY
4	38496061	Doka
5	126701222	LADOS
7	99958075	MRM Manlywear
8	176250783	MRM Manlywear
9	99968881	MRM Manlywear
10	31637590	MRM Manlywear
12	215072155	OEM

Hình 13: Hiển thị lại DataFrame không chứa dòng trùng lặp

Xử lý các giá trị NaN.

```
# Xử lý giá trị NaN ("Root")
df <- df[!(df$categories == "Root"), ]
df
```

2.3 Thống kê mô tả

Thống kê mô tả tập dữ liệu.

```
summary(df)
```

Kết quả:

```
      id      brand_name      categories
Min.   : 4027231  Length:343      Length:343
1st Qu.: 75325078  Class :character  Class :character
Median :124748904  Mode  :character  Mode  :character
Mean    :136813678
3rd Qu.:195991159
Max.    :273365715

      discount      discount_rate      list_price
Min.   :      0  Min.   : 0.00  Min.   : 17000
1st Qu.:      0  1st Qu.: 0.00  1st Qu.: 119000
Median : 13000  Median :11.00  Median : 215000
Mean    : 66700  Mean    :18.23  Mean    : 282051
3rd Qu.: 99500  3rd Qu.:35.00  3rd Qu.: 352500
Max.    :899000  Max.    :66.00  Max.    :2090000

      price      rating_average      review_count
Min.   : 17000  Min.   :0.000  Min.   :  0.0
1st Qu.: 99000  1st Qu.:4.200  1st Qu.:  3.0
Median :168800  Median :4.600  Median : 18.0
Mean    :215350  Mean    :4.067  Mean    :101.7
3rd Qu.:269000  3rd Qu.:4.800  3rd Qu.: 96.0
Max.    :2090000  Max.    :5.000  Max.    :4772.0
```

Hình 14: Thống kê mô tả tập dữ liệu

- **id:** Cột chứa các giá trị id. Các giá trị bao gồm giá trị tối thiểu (Min.), giá trị ở phân vị 25% (1st Qu.), giá trị trung vị (Median), giá trị trung bình (Mean), giá trị ở phân vị 75% (3rd Qu.), và giá trị tối đa (Max.).
- **brand_name:** Cột chứa thông tin về tên thương hiệu (brand_name). Đây có vẻ là kiểu dữ liệu ký tự (character). Không có thống kê mô tả nào được hiển thị vì đây là biến phân loại.
- **categories:** Cột chứa thông tin về loại sản phẩm (categories). Cũng là kiểu dữ liệu ký tự (character) và không có thống kê mô tả.
- **discount:** Cột chứa giá trị giảm giá. Các thống kê mô tả bao gồm giá trị tối thiểu, phân vị 25%, trung vị, trung bình, phân vị 75%, và giá trị tối đa.
- **discount_rate:** Cột chứa tỷ lệ giảm giá. Thống kê mô tả tương tự như cột discount.
- **list_price:** Cột chứa giá niêm yết. Thống kê mô tả bao gồm giá trị tối thiểu, phân vị 25%, trung vị, trung bình, phân vị 75%, và giá trị tối đa.

- **price:** Cột chứa giá bán. Thống kê mô tả giống như cột list_price.
- **rating_average:** Cột chứa giá trị trung bình đánh giá. Thống kê mô tả bao gồm giá trị tối thiểu, phân vị 25%, trung vị, trung bình, phân vị 75%, và giá trị tối đa.
- **review_count:** Cột chứa số lượng đánh giá. Thống kê mô tả bao gồm giá trị tối thiểu, phân vị 25%, trung vị, trung bình, phân vị 75%, và giá trị tối đa.

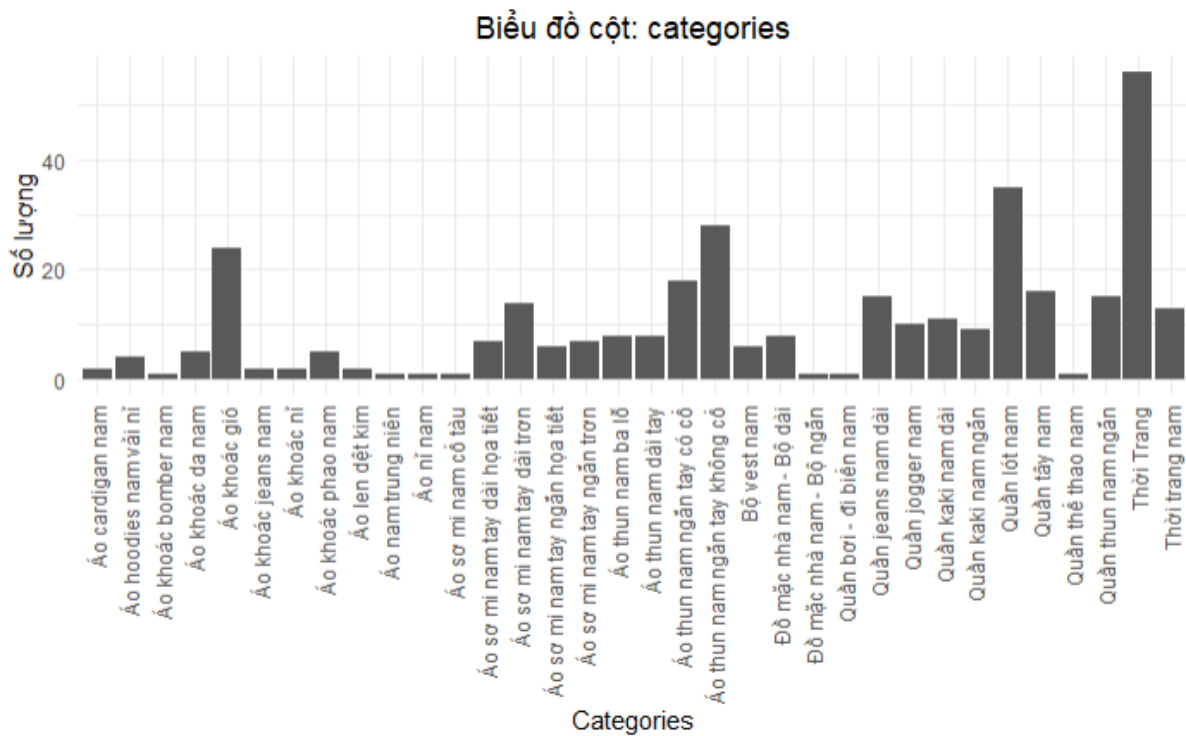
2.4 Trực quan hóa dữ liệu

2.4.1 Biểu đồ cột

Về biểu đồ cột.

```
ggplot(df, aes(x = categories)) +
  geom_bar() +
  labs(title = "Biểu đồ cột dọc cho categories", x = "Categories", y = "Số
lượng") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
    plot.title = element_text(hjust = 0.5) # Đặt title ở giữa
  )
```

Kết quả:



Hình 15: Biểu đồ cột

Biểu đồ này thể hiện số lượng của từng loại trong danh mục (categories).

2.4.2 Biểu đồ tròn

Phân loại thành các biến thành áo, quần và khác.

```
# Lọc dữ liệu theo yêu cầu
ao <- nrow(df %>%filter(grepl("áo", categories, ignore.case = TRUE)))
quan <- nrow(df %>%filter(grepl("quần", categories, ignore.case = TRUE)))
khac <- nrow(df %>%filter(!grepl("áo", categories, ignore.case = TRUE) &
!grepl("quần", categories, ignore.case = TRUE)))
```

Về biểu đồ tròn thể hiện tỉ lệ phân loại của categories.

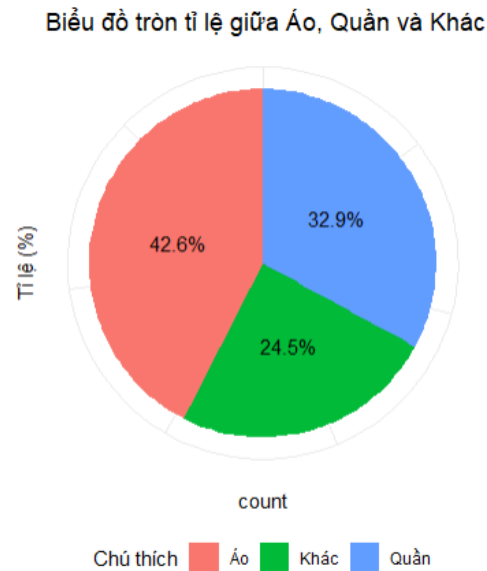
```
data <- data.frame(
  category = c("Áo", "Quần", "Khác"),
  count = c(ao, quan, khac)
)

# Tính tổng số lượng
total_count <- sum(data$count)

# Tính phần trăm
data$percentage <- (data$count / total_count) * 100

# Biểu đồ tròn thể hiện tỉ lệ và thêm phần trăm, xóa số 100, 200, 300 ngoài
vòng tròn
ggplot(data, aes(x = "", y = count, fill = category)) +
  geom_bar(stat = "identity", width = 1) +
  geom_text(aes(label = sprintf("%.1f%%", percentage)), position =
position_stack(vjust = 0.5)) +
  coord_polar("y") +
  labs(title = "Biểu đồ tròn tỉ lệ giữa Áo, Quần và Khác", fill = "Chú thích",
x = "Tỉ lệ (%)") +
  theme_minimal() +
  theme(legend.position = "bottom", axis.text = element_blank())
```

Kết quả:



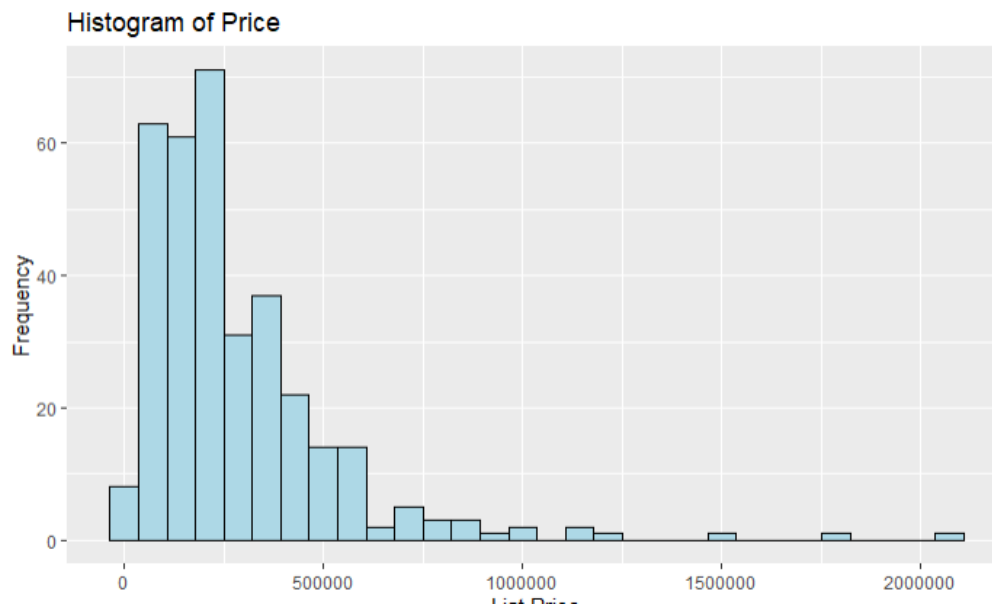
Hình 16: Biểu đồ tròn

2.4.2 Biểu đồ histogram

Vẽ biểu đồ histogram dựa trên mật độ phân phối của giá sản phẩm.

```
ggplot(df, aes(x = list_price)) +  
  geom_histogram(bins = 30, fill = "lightblue", color = "black") +  
  labs(title = "Histogram of Price", x = "List Price", y = "Frequency")
```

Kết quả:



Hình 17: Biểu đồ histogram

CHƯƠNG 3: CHỨNG MINH VÀ PHÂN TÍCH CÁC GIẢ THUYẾT, BÀI TOÁN

3.1 Chứng minh giả thuyết (Hypothesis Testing)

Để chứng minh giả thuyết (hypothesis testing) cho một biến cụ thể trong tập dữ liệu, ta cần xác định giả thuyết null (H_0) và giả thuyết chính (H_1), sau đó thực hiện kiểm định thống kê để đưa ra quyết định về việc bác bỏ hay không bác bỏ giả thuyết null.

Dưới đây là một ví dụ giả định chứng minh sự khác biệt về giá trị trung bình của cột price giữa hai nhóm sản phẩm có brand_name khác nhau (OEM và LADOS).

Bước 1: Xác định giả thuyết.

- Giả thuyết null (H_0): Giá trị trung bình của price đối với brand_name OEM bằng giá trị trung bình của price đối với brand_name LADOS.
- Giả thuyết chính (H_1): Giá trị trung bình của price đối với brand_name OEM không bằng giá trị trung bình của price đối với brand_name LADOS.

Bước 2: Thực hiện kiểm định t.

Trong đoạn mã bên dưới, ta giả định rằng giá trị của cột price phụ thuộc vào brand_name. Sau đó, tôi thực hiện kiểm định t để kiểm tra xem có sự khác biệt đáng kể về giá trị trung bình của price giữa hai nhóm OEM và LADOS hay không.

```
# Kiểm định t về sự khác biệt giữa giá trị trung bình của price đối với OEM và LADOS
t_test_result <- t.test(df$price[df$brand_name == "OEM"],
df$price[df$brand_name == "LADOS"])

# Hiển thị kết quả
print(t_test_result)
```

Kết quả:

```
welch Two Sample t-test

data: df$price[df$brand_name == "OEM"] and df$price[df$brand_name == "LADOS"]
t = -1.2622, df = 25.291, p-value = 0.2184
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -101077.17  24233.45
sample estimates:
mean of x mean of y
169039.7  207461.5
```

Hình 18: Kết quả kiểm định t

- **t = -1.2622** (Giá trị thống kê t): Nếu giá trị này gần 0, có nghĩa là không có sự khác biệt đáng kể giữa hai nhóm. Trong trường hợp này, giá trị t là âm, cho biết trung bình của nhóm "OEM" có thể nhỏ hơn so với nhóm "LADOS".
- **df = 25.291** (Độ tự do của phân phối t): Đây là một ước lượng dựa trên dữ liệu và được sử dụng để xác định giá trị p.
- **p-value = 0.2184**: Đây là xác suất để thấy giá trị t thực tế hoặc lớn hơn, giả sử giả thuyết null là đúng (không có sự khác biệt giữa các nhóm). Nếu giá trị p lớn, chúng ta không thể bác bỏ giả thuyết null. Ở đây, giá trị p lớn (0.2184), nên không có đủ bằng chứng để bác bỏ giả thuyết null.
- **alternative hypothesis: true difference in means is not equal to 0** (Giả thuyết thay thế): Sự khác biệt thực sự giữa trung bình không bằng 0. Điều này có nghĩa là kiểm định xem có sự khác biệt nào đó giữa hai nhóm hay không.
- **95 percent confidence interval: -101077.17 to 24233.45** (Khoảng tin cậy 95%): Nếu giá trị 0 nằm trong khoảng này, chúng ta không thể kết luận rằng có sự khác biệt đáng kể. Trong trường hợp này, khoảng tin cậy chứa giá trị 0, cũng hỗ trợ cho việc không có sự khác biệt đáng kể.
- **sample estimates: mean of x mean of y: 169039.7 207461.5**: Ước lượng trung bình của nhóm "OEM" là 169039.7 và của nhóm "LADOS" là 207461.5.

Tóm lại, dựa vào giá trị p lớn (0.2184) và khoảng tin cậy chứa giá trị 0, không có đủ bằng chứng để bác bỏ giả thuyết null, và ta không có đủ lý do để tin rằng có sự khác biệt đáng kể giữa trung bình của hai nhóm "OEM" và "LADOS".

Bước 3: Kiểm tra giả thuyết.

Kết quả của kiểm định t sẽ cung cấp giá trị p (p-value). Nếu giá trị p nhỏ hơn một ngưỡng ý nghĩa (thường là 0.05), ta có thể bác bỏ giả thuyết null và chấp nhận giả thuyết chính, kết luận rằng có sự khác biệt đáng kể giữa hai nhóm.

```
# Kiểm tra giả thuyết null
if (t_test_result$p.value < 0.05) {
  print("Bác bỏ giả thuyết null: Có sự khác biệt đáng kể giữa OEM và LADOS")
} else {
  print("Chấp nhận giả thuyết null: Không có sự khác biệt đáng kể giữa OEM và LADOS")
}
```


Kết quả:

```
[1] "chấp nhận giả thuyết null: không có sự khác biệt đáng kể giữa OEM và LADOS"
```

Hình 19: Kết quả kiểm tra giả thuyết

3.2 Dự đoán

3.2.1 Linear Regression

```
# Mô hình hồi quy tuyến tính
model_lm <- lm(price ~ discount + discount_rate + list_price + rating_average,
data = df)

# Hiển thị tóm tắt mô hình
summary(model_lm)
```

Kết quả:

```
call:
lm(formula = price ~ discount + discount_rate + list_price +
    rating_average, data = df)

Residuals:
    Min       1Q   Median       3Q      Max
-4.890e-10 -4.960e-11 -4.010e-11 -2.900e-11  1.183e-08

Coefficients:
              Estimate Std. Error    t value Pr(>|t|)
(Intercept)  4.023e-10  1.221e-10  3.294e+00  0.00109 **
discount     -1.000e+00  6.411e-16 -1.560e+15 < 2e-16 ***
discount_rate 1.932e-12  2.844e-12  6.790e-01  0.49747
list_price    1.000e+00  1.990e-16  5.025e+15 < 2e-16 ***
rating_average 1.233e-11  2.461e-11  5.010e-01  0.61662
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.457e-10 on 338 degrees of freedom
Multiple R-squared:  1,    Adjusted R-squared:  1
F-statistic: 7.877e+30 on 4 and 338 DF,  p-value: < 2.2e-16
```

Hình 20: Tóm tắt mô hình Linear Regression

1. Residuals (Dư thừa):

- **Min:** Giá trị dư thừa nhỏ nhất.
- **1Q (Phân vị 25%):** Khoảng 25% dữ liệu có giá trị dư thừa nhỏ hơn giá trị này.
- **Median (Trung vị):** Giá trị dư thừa tại vị trí trung vị (50% dữ liệu có giá trị dư thừa nhỏ hơn).
- **3Q (Phân vị 75%):** Khoảng 75% dữ liệu có giá trị dư thừa nhỏ hơn giá trị này.
- **Max:** Giá trị dư thừa lớn nhất.

2. Coefficients (Hệ số):

- **Estimate:** Ước lượng của hệ số tương ứng với mỗi biến độc lập.
 - **Std. Error (Độ lệch chuẩn):** Độ lệch chuẩn của ước lượng.
 - **t value (Giá trị t):** Giá trị thống kê t.
 - **Pr(>|t|) (Giá trị p):** Giá trị p, xác suất của giá trị t hiện tại hoặc lớn hơn nếu giả thuyết null là đúng.
3. **Residual standard error (Sai số tiêu chuẩn của dư thừa):** Độ lệch chuẩn của giá trị dư thừa, đo lường sự biến động của các điểm dữ liệu quanh đường hồi quy.
4. **Multiple R-squared (R-quadrat đa biến):** Đo lường mức độ giải thích bởi mô hình. Giá trị 1 có nghĩa là mô hình giải thích toàn bộ biến động của phản hồi.
5. **Adjusted R-squared (R-quadrat được điều chỉnh):** Đo lường mức độ giải thích điều chỉnh cho số lượng biến động trong mô hình.
6. **F-statistic (Thống kê F):** Kiểm định F cho biết xem có sự khác biệt đáng kể giữa mô hình được fit và mô hình không có biến động nào không.
7. **p-value (Giá trị p):** Xác suất để thấy giá trị F hiện tại hoặc lớn hơn nếu giả thuyết null là đúng.

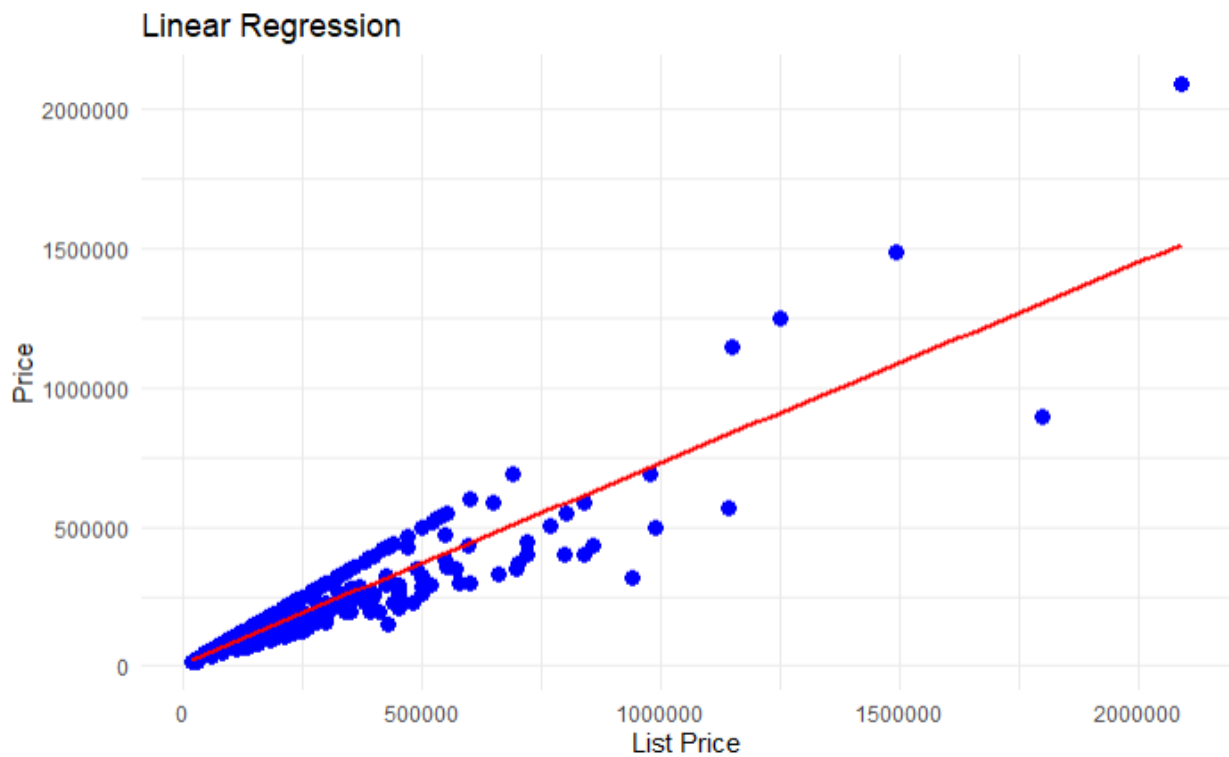
Tóm lại:

- Mô hình có vẻ rất phù hợp với dữ liệu, với R-quadrat đa biến và R-quadrat được điều chỉnh đều bằng 1.
- Các hệ số của các biến như "discount", "discount_rate", "list_price", "rating_average" được ước lượng.
- Các giá trị p cho tất cả các hệ số đều rất nhỏ, đặc biệt là cho "discount", "list_price" (p-value < 2.2e-16), có nghĩa là chúng có ảnh hưởng đáng kể đến biến phụ thuộc "price".
- F-statistic là một giá trị rất lớn và giá trị p của nó cũng rất nhỏ, có nghĩa là mô hình là đủ tốt để giải thích dữ liệu.

Vẽ biểu đồ trực quan.

```
# Vẽ biểu đồ scatterplot với đường hồi quy
library(ggplot2)
ggplot(df, aes(x = list_price, y = price)) +
  geom_point(color = "blue", size = 3) +
  geom_smooth(method = "lm", se = FALSE, color = "red", formula = y ~ x) +
  labs(title = "Linear Regression", x = "List Price", y = "Price") +
  theme_minimal()
```

Kết quả:



Hình 21: Biểu đồ Linear Regression

3.2.2 K-Means

Phân loại dữ liệu thành 3 cụm dựa trên các biến "discount", "discount_rate", "list_price", và "rating_average".

```
# K-Means Clustering
model_kmeans <- kmeans(df[, c("discount", "discount_rate", "list_price",
"rating_average")], centers = 3)

# Thêm thông tin phân cụm vào dữ liệu
df$cluster <- as.factor(model_kmeans$cluster)
```

Kết quả:

K-means clustering with 3 clusters of sizes 19, 107, 217

Cluster means:

	discount	discount_rate	list_price	rating_average
1	324500.00	35.26316	1029526.3	3.000000
2	110310.00	26.42991	415719.6	4.132710
3	22624.79	12.70046	150693.1	4.128111

Clustering vector:

```
1 2 3 4 5 7 8 9 12 13 14 16 17 18 19 20 21 22 23 25 27 28 29 30 31 33 34 35 38 39 40 41 44 45 46 47 48 49 50
2 3 1 3 2 3 3 2 2 3 2 3 2 3 3 3 3 3 2 3 3 2 3 3 2 2 3 3 3 2 3 3 3 2 2 2 2 2
52 53 54 55 62 63 64 65 66 69 71 72 73 74 75 76 77 79 80 81 83 84 85 86 87 88 90 91 92 93 94 95 96 98 99 100 101 102 103
3 3 1 3 3 3 1 3 3 3 3 2 3 3 3 2 3 2 3 2 3 2 3 3 2 3 2 3 3 3 3 3 3 3 3 3 3 3
104 105 107 108 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 131 132 133 134 135 137 139 140 142 143 144 145 146 147 148
2 3 3 3 3 2 2 3 1 3 3 2 3 3 3 2 3 2 2 2 3 3 1 1 1 3 2 3 2 3 2 2 3 3 3 2 3 3 3
149 150 151 152 153 154 155 156 157 158 159 160 161 162 164 165 166 167 169 170 172 175 176 177 178 179 180 181 182 184 185 188 189 190 191 192 193 194 195
2 3 3 2 2 3 2 3 3 3 3 3 3 3 3 2 2 1 3 3 3 3 3 3 3 2 2 3 3 3 3 3 3 3 3 2 2 3 3
197 198 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 232 233 234 235 236 238
3 3 2 2 2 3 2 3 2 3 2 3 2 3 2 2 3 3 3 3 3 3 3 3 3 2 3 3 1 2 2 2 1 3 3 2 3 3 3
239 240 241 243 244 245 246 247 249 250 251 252 253 254 255 256 257 258 259 261 262 265 266 267 268 269 270 271 272 273 275 276 277 278 279 281 282 283 284
3 3 2 2 3 3 3 3 3 3 2 3 2 3 3 3 3 2 2 3 2 2 3 3 3 1 1 1 3 3 3 2 2 3 3 2 1 3 3
285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 301 303 304 305 306 307 308 309 312 313 314 315 316 317 318 322 323 324 325 326 327 328 329 330
3 3 3 3 3 2 2 3 3 3 3 3 3 3 3 3 3 2 2 3 2 2 3 3 3 1 1 1 3 3 3 2 2 3 3 2 2 3 3
331 332 334 335 336 337 338 340 341 343 346 347 348 350 352 353 354 355 356 359 360 361 362 363 364 365 366 368 369 370 372 373 374 375 379 380 382 384 387
3 3 3 2 3 3 3 2 3 3 3 3 3 3 3 2 1 3 3 3 1 3 3 3 2 3 2 2 3 1 3 3 2 3 2 3 2 2
389 391 392 393 394 396 397 398 399 400 403 404 407 408 409 411 412 413 414 416 417 418 419 422 423 424 425 426 427 429 430
3 2 3 2 1 1 3 3 2 3 2 3 3 2 2 3 3 3 3 2 2 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```

within cluster sum of squares by cluster:
[1] 3.680026e+12 1.863366e+12 1.239664e+12
(between_SS / total_SS = 72.8 %)

Available components:

[1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss" "size" "iter" "ifault"

Hình 22: Thông tin phân cụm dữ liệu

1) Cluster means (Trung tâm cụm):

a) Cluster 1:

- i) discount: 324,500.00
- ii) discount_rate: 35.26
- iii) list_price: 1,029,526.3
- iv) rating_average: 3.00

b) Cluster 2:

- i) discount: 110,310.00
- ii) discount_rate: 26.43
- iii) list_price: 415,719.6
- iv) rating_average: 4.13

c) Cluster 3:

- i) discount: 22,624.79
- ii) discount_rate: 12.70
- iii) list_price: 150,693.1
- iv) rating_average: 4.13

2) Clustering vector (Vector phân cụm):

- a) Liệt kê cụm mà mỗi quan sát thuộc về. Ví dụ, quan sát thứ 1 thuộc về cụm 2, quan sát thứ 2 thuộc về cụm 3, và tiếp tục.

3) Within cluster sum of squares by cluster (Tổng bình phương trong cụm):

- a) Cho biết tổng bình phương của khoảng cách từ mỗi điểm đến trung tâm của cụm nó thuộc về. Giảm giá trị này là mục tiêu của việc cải thiện mô hình.

4) Percentage of variance explained (Tỉ lệ phương sai được giải thích):

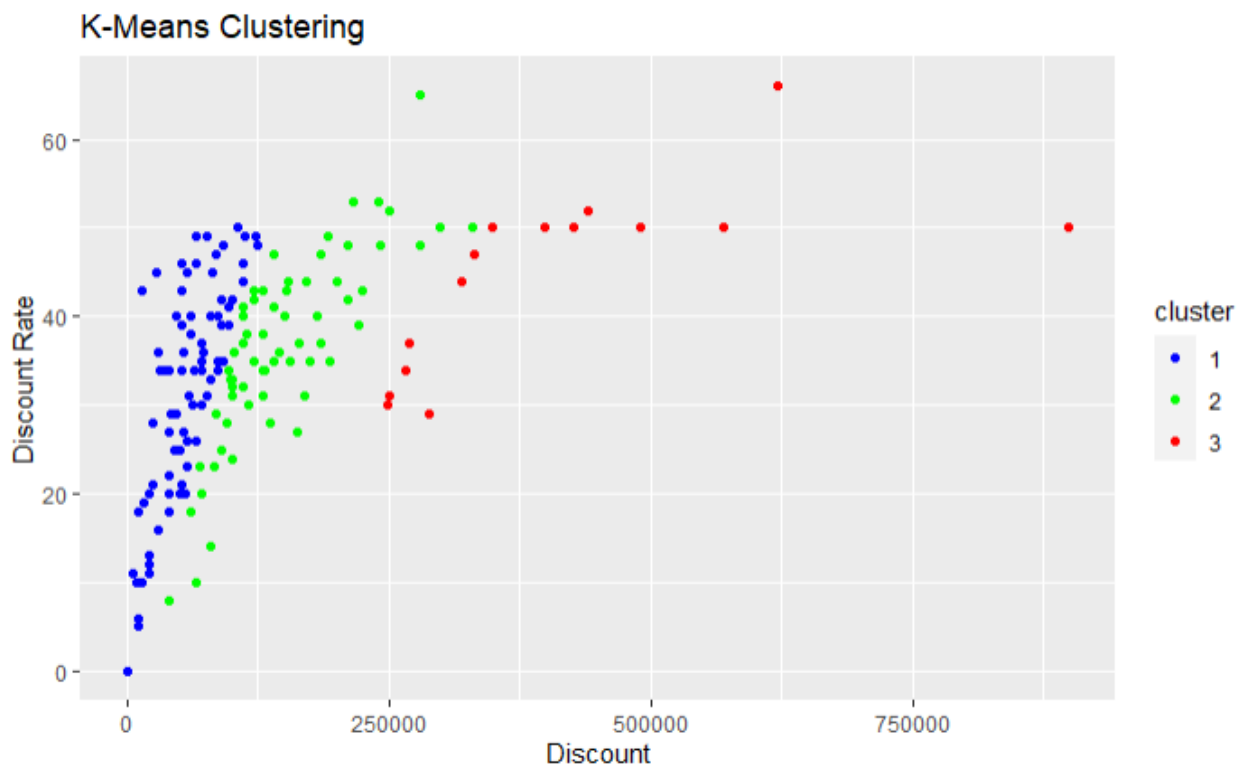
- a) Tỉ lệ phương sai được giải thích bởi phương sai giữa các cụm so với tổng phương sai.

Trong trường hợp này, dữ liệu đã được phân thành 3 cụm, và phương sai giữa các cụm là khoảng 72.8% của tổng phương sai. Kết quả này cung cấp cái nhìn tổng quan về cách dữ liệu đã được phân loại và cụm trung tâm của từng nhóm.

Vẽ biểu đồ.

```
# Vẽ biểu đồ scatterplot
library(ggplot2)
ggplot(df, aes(x = discount, y = discount_rate, color = cluster)) +
  geom_point() +
  labs(title = "K-Means Clustering", x = "Discount", y = "Discount Rate") +
  scale_color_manual(values = c("blue", "green", "red"))
```

Kết quả:



Hình 23: Biểu đồ KMeans

3.3 Phân tích PCA

Phân tích thành phần chính (PCA) là một phương pháp giảm chiều dữ liệu, giúp tìm ra các biến quan trọng nhất và giảm số lượng biến để dễ dàng hiểu và trực quan hóa dữ liệu. Dưới đây là một hướng dẫn sơ bộ về cách thực hiện PCA dữ liệu.

Bước 1: Chuẩn bị dữ liệu.

```
# Lựa chọn các biến số cần tham gia vào PCA
df_pca <- df[, c("discount", "list_price", "price")]

# Chuẩn bị dữ liệu cho PCA (loại bỏ các cột không phải dạng số và xử lý missing data)
df_pca <- na.omit(df_pca)
df_pca_scaled <- scale(df_pca) # Chuẩn hóa dữ liệu
```

Bước 2: Thực hiện PCA.

```
# Thực hiện PCA
pca_result <- prcomp(df_pca_scaled)

# Hiển thị kết quả
print(summary(pca_result))
```

Kết quả:

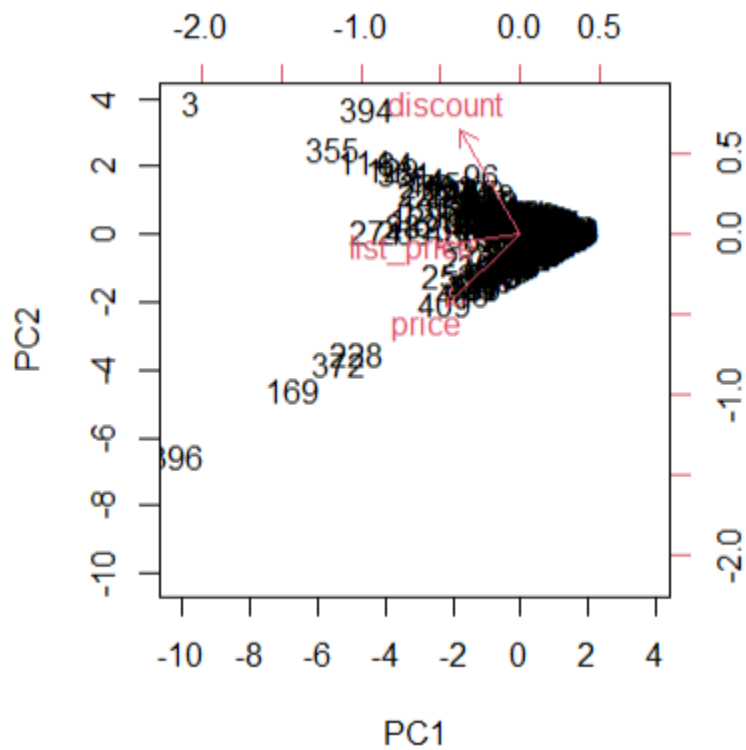
```
Importance of components:
              PC1      PC2      PC3
Standard deviation   1.5029 0.8610 7.379e-16
Proportion of Variance 0.7529 0.2471 0.000e+00
Cumulative Proportion 0.7529 1.0000 1.000e+00
```

Hình 24: Kết quả PCA

Bước 3: Trực quan hóa kết quả PCA.

```
# Biểu đồ các thành phần chính
biplot(pca_result, scale = 0)
```

Kết quả:



Hình 25: Biểu đồ trực quan PCA

LINK MÃ NGUỒN

Link Github: [minhkhanh-coder/Tiki_CrawlData \(github.com\)](https://github.com/minhkhanh-coder/Tiki_CrawlData)

TÀI LIỆU THAM KHẢO

KẾT QUẢ KIỂM TRA ĐẠO VĂN

❧ ❧

(Ký và ghi rõ họ tên)