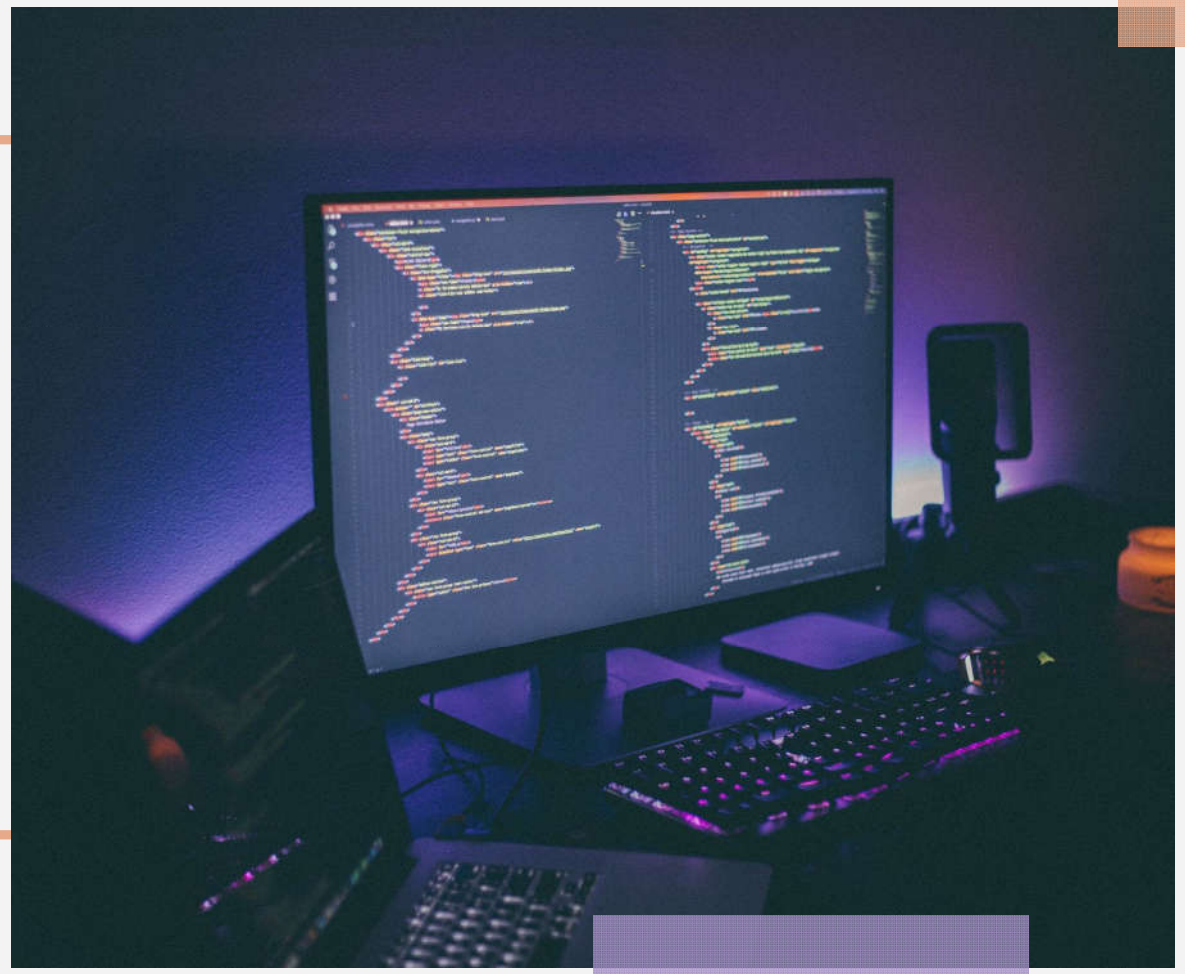
Web scraping project: Lululemon

Maggie Han

Feb 22, 2023



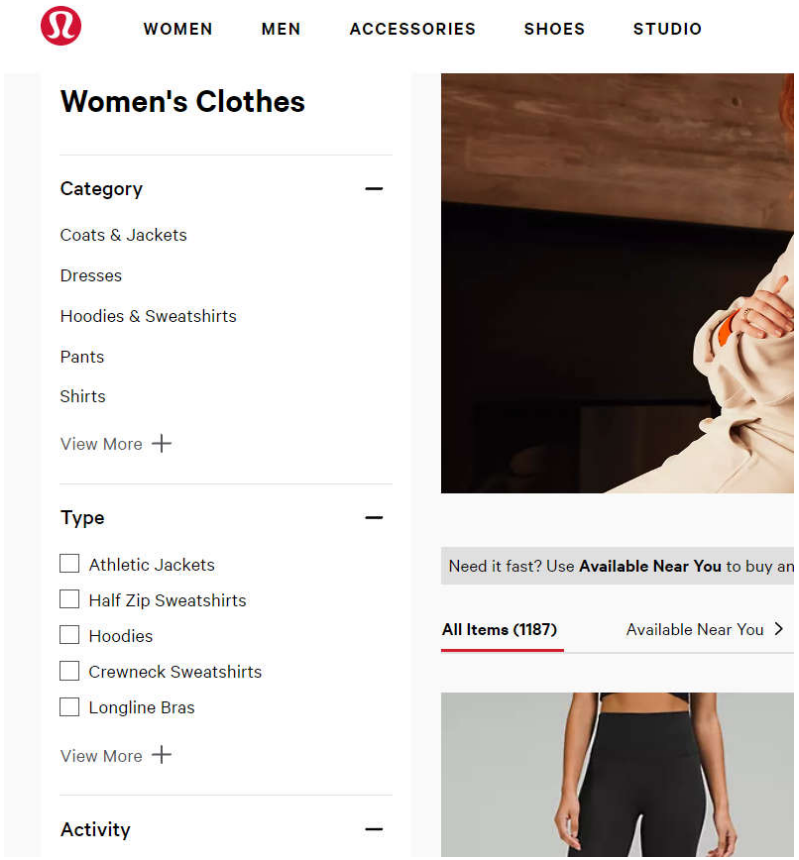
CONTENTS

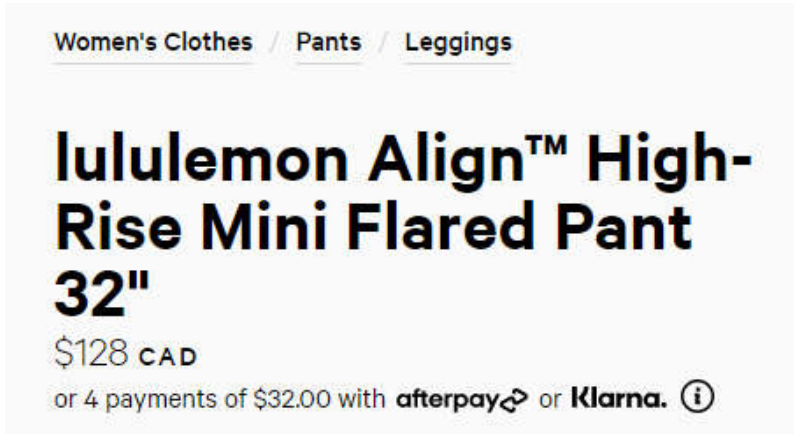
1. Introduction

- Type: Public
- Industry: Retail
- Products: Sportswear
- Founded: 1998;
- Founder: Chip Wilson
- Revenue: US$6.26 billion (2021)
- Number of locations: 574 (Dec 2021)
- Headquarter:
  - Vancouver, British Columbia, Canada
- Website: lululemon.com

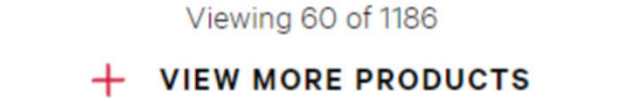
Lululemon Opened 3-Level Flagship Store at Yonge and Bloor Intersection in Downtown Toronto

2. Web scraping process





3. Challenges



Steps:

1. Continue scrolling page down to find all the products
2. Need to click button to explore more
3. At the end the button is no more exist, break the while loop and scroll down to the page bottom

```
scroll_pause_time = 2 # You can set your own pause time.
screen_height = driver.execute_script("return window.screen.height;")   # get the screen height of the web
i = 1

while True:
    # scroll one screen height each time
    driver.execute_script("window.scrollTo(0, {screen_height}*{i});".format(screen_height=screen_height, i=i))
    i += 1
    sleep(scroll_pause_time)
    # update scroll height each time after scrolled, as the scroll height can change after we scrolled the page
    scroll_height = driver.execute_script("return document.body.scrollHeight;")

    # Parse the HTML using BeautifulSoup
    soup = BeautifulSoup(driver.page_source, 'html.parser')
    # Find the button element
    button = soup.find('button', {'class': 'iconButtonIcon-3D21Q'})
    # Scroll to the button element using Selenium
    try:
        button_element = driver.find_element(By.XPATH, '//*[@id="main-content"]/div/section/div/div[2]/button/div/span')
        # Click the button
        button_element.click()
    except:
        #button not exist
        driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
        break
    # Break the loop when the height we need to scroll to is larger than the total scroll height
    if (screen_height) * i > scroll_height:
        break
```

3. Challenges

```
▼<li data-testid="breadcrumb-li" data-slash="/" class="breadcrumb-1t2hi">
   <a class="link OneLinkTx" href="/story/women">Women's Clothes</a> == $0
   ::after
 </li>
▼<li data-testid="breadcrumb-li" data-slash="/" class="breadcrumb-1t2hi">
   <a class="link OneLinkTx" href="/c/women-pants/_/N-8r2">Pants</a>
   ::after
 </li>
▼<li data-testid="breadcrumb-li" class="breadcrumb-1t2hi">
   <a class="link OneLinkTx" href="/c/womens-sweatpants/_/N-8r5">Sweatpants<
   ::after
 </li>
```

1. Three headings share the same class
2. Use for loop to find all headers and put them in different columns
3. Use if condition to input NA for products that have no subcategories

```
driver.get(link)
soup = BeautifulSoup(driver.page_source,'html.parser')
# as gender, catagory and sub_cata share the same class, using a for loop to find all
i = 0
try:
    for header in soup.find('ul',class_='breadcrumbs-FTRV6 breadcrumbs').find_all('li'):
        if i == 0 :
            info['gender'].append(header.get_text())
            i += 1
            print(header.get_text())
        elif i == 1:
            info['catagory'].append(header.get_text())
            i += 1
            print(header.get_text())
        else:
            info['sub_cata'].append(header.get_text())
            print(header.get_text())

    # for products with no catagory and sub_cata
    if len(soup.find('ul',class_='breadcrumbs-FTRV6 breadcrumbs').find_all('li'))== 1 :
        info['catagory'].append('NA')
        print('NA')
        info['sub_cata'].append('NA')
        print('NA')

    # for products with no sub_cata
    if len(soup.find('ul',class_='breadcrumbs-FTRV6 breadcrumbs').find_all('li'))== 2 :
        info['sub_cata'].append('NA')
        print('NA')

# for links may not exist anymore and being updated during the scarping process
except:
    info['gender'].append('NA')
    print('NA')
    info['catagory'].append('NA')
    print('NA')
    info['sub_cata'].append('NA')
    print('NA')
```

3. Challenges

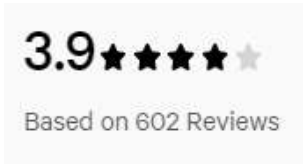
```
# setup driver for finding review and num_of_reviews, as they don't have class and cannot be found using beautiful soup
# move the page down to allow review present in the screen to avoid 'no such element expectation' error
driver.execute_script("window.scrollTo(0, 2200)")
sleep(2)

# collect review point
try:
    review = driver.find_element(By.XPATH,'//*[@id="main-content"]/div[1]/section/div[4]/div/div/div/div/div/div/div[1]/
    info['reviews'].append(review)
    print(review)
except:
    info['reviews'].append('NA')
    print('NA')

# collect number of reviews
try:
    num = driver.find_element(By.XPATH, '//*[@id="main-content"]/div[1]/section/div[4]/div/div/div/div/div/div/div[1]/di
    info['num_of_review'].append(num)
    print(num)
except:
    info['num_of_review'].append('NA')
    print('NA')
```

Scroll down to let review appear in the screen

Using selenium to find XPATH of review and num_of_review




```
<h2 class="reviews-header_reviewTitle__E1RU4 lll-text-xlarge" data-testid="review-
header-title">Reviews</h2>
<div aria-label="Average Rating - 4.1 out of 5 stars. Based on 478 Reviews" class=
"reviews-header_reviewAvgRatingContainer__yA2_j" data-testid="review-header-avg-ra
ting-container">
 <span aria-label="4.1 out of 5 stars" class="lll-font-weight-medium reviews-head
  er_reviewsRatingLabel__dGx5H lll-text-medium">
    <span aria-hidden="true">4.1</span> == $0
  </span>
```

4. Data clean and analysis

- Convert str type to int in review_num column

```
#make a copy of dateframe
women = womens
men = mens
```

```
# extract number
# fill na to 0
# convert object to int
women['review_num']= women['review_num'].str.extract('(\d+)').fillna(0).astype(int)
women.info()
```

price	review	review_num
$128 CAD	4.2	Based on 490 Reviews
$68 CAD	4.3	Based on 226 Reviews
$138 CAD	4.3	Based on 14 Reviews
$198 CAD	4.4	Based on 18 Reviews
$148 CAD	5.0	Based on 2 Reviews

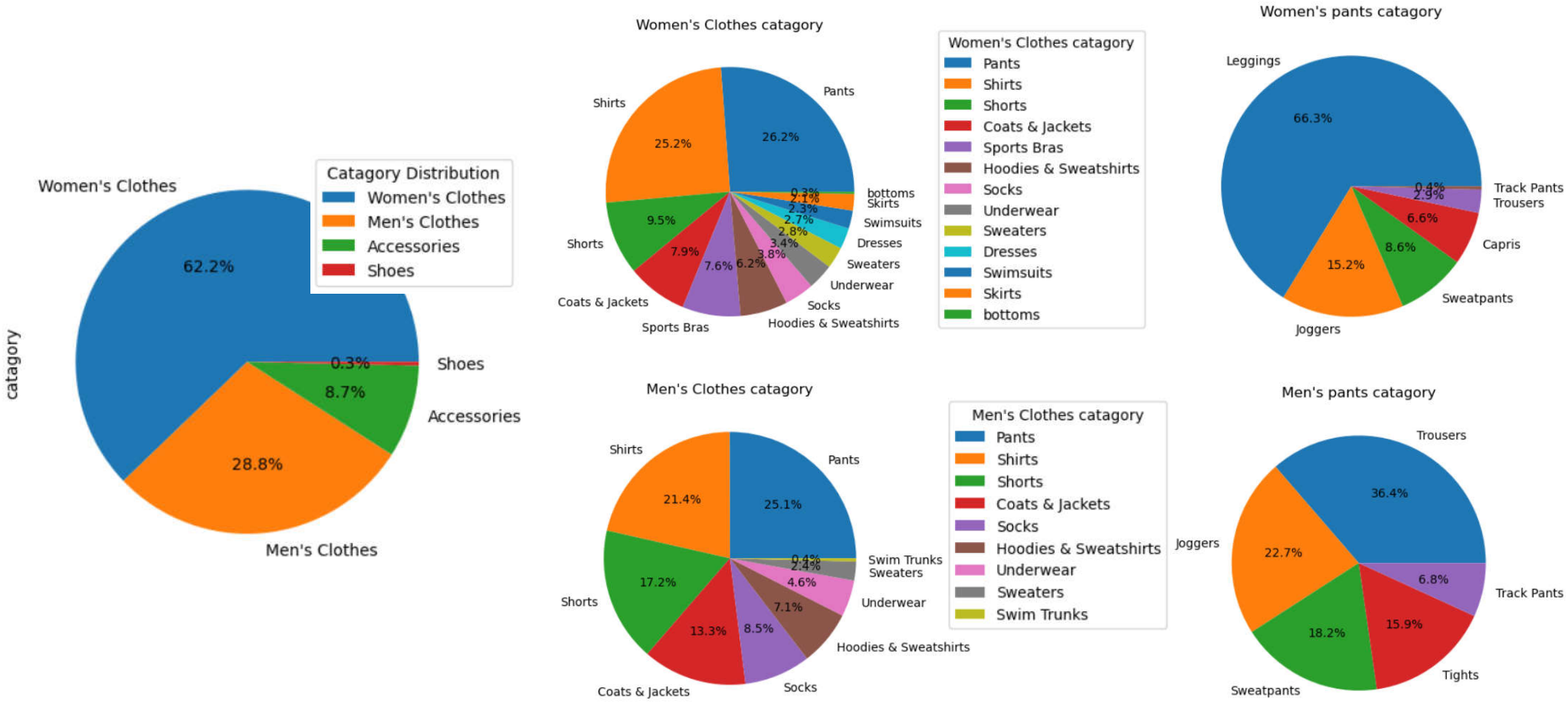
- Convert str type to float in price column

```
men['price'] = men['price'].str.extract('(\d+\.\d+|\d+)').astype(float)
men.head()
```

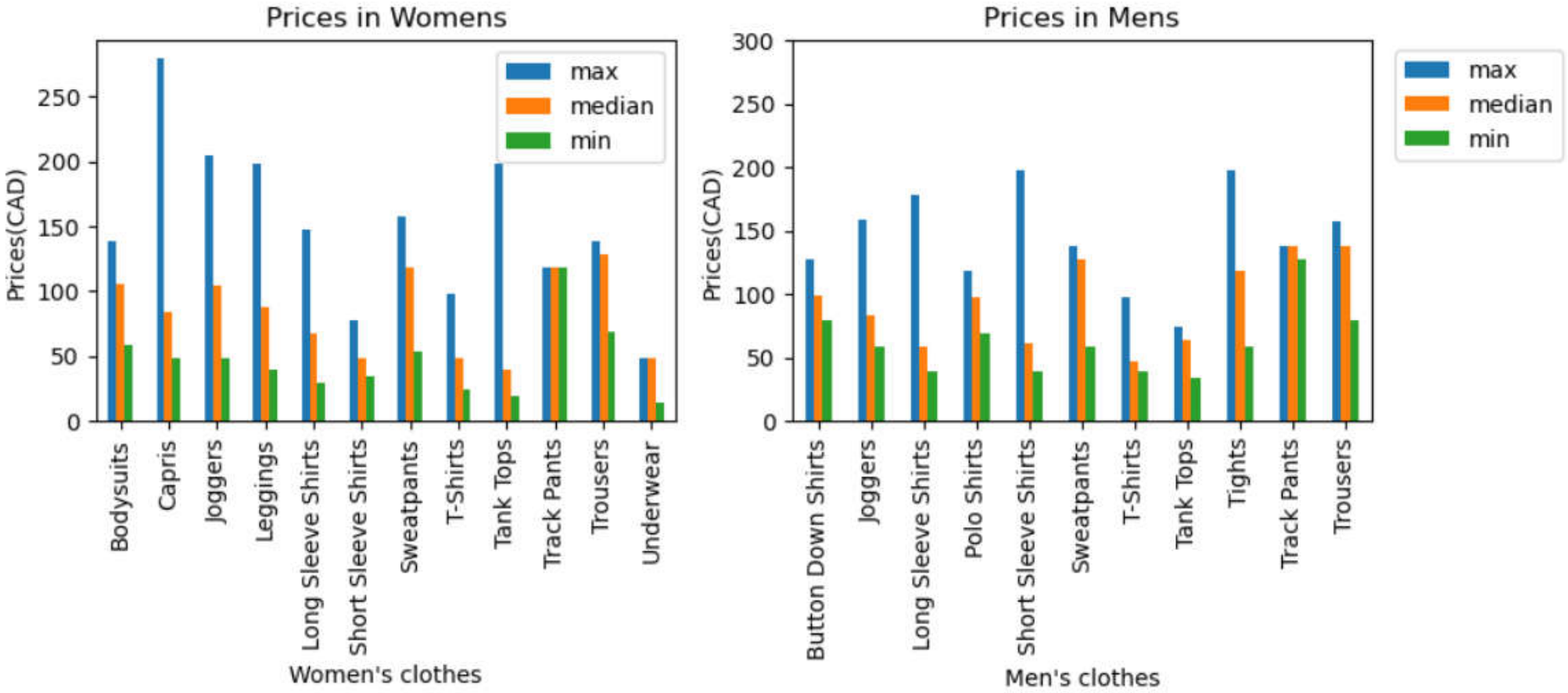
	catagory	sub_cata_1	sub_cata_2	product	price	review	review_num
0	Men's Clothes	Hoodies & Sweatshirts	NaN	City Sweat Pullover Hoodie	128.0	4.2	490
1	Men's Clothes	Shorts	NaN	City Sweat Short 9" Online Only	68.0	4.3	226
2	Men's Clothes	Pants	NaN	Utilitech Pull-On Relaxed-Fit Pant Online Only	138.0	4.3	14
3	Men's Clothes	Coats & Jackets	NaN	Switch Over Bomber Jacket Cotton Blend	198.0	4.4	18
4	Men's Clothes	Hoodies & Sweatshirts	NaN	French Terry Oversized Half Zip	148.0	5.0	2

- Concat two tables into one and remove duplicates

Category distribution



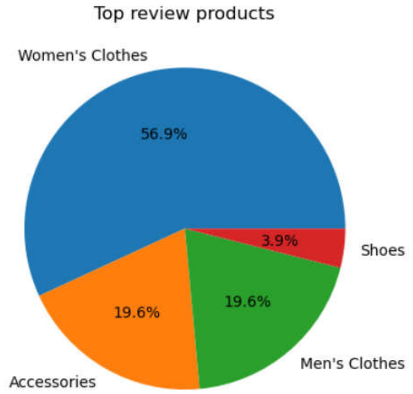
Prices in Mens and Womens

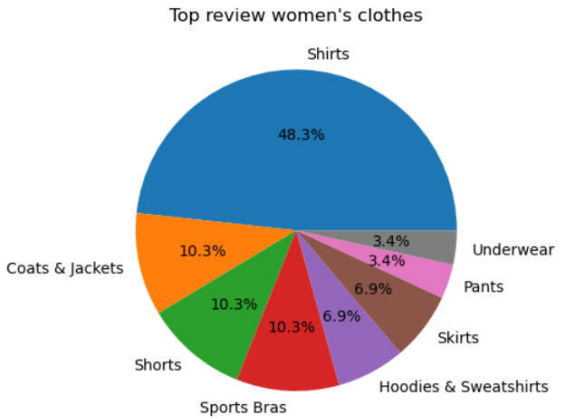


5. Interesting findings

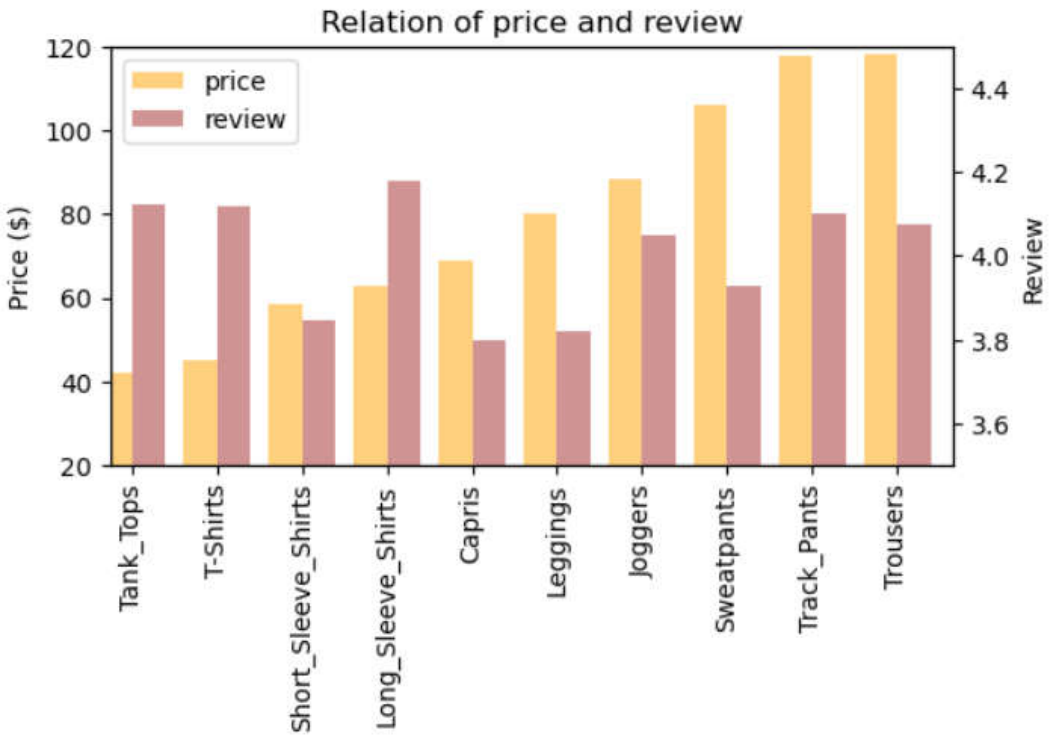
top_review =

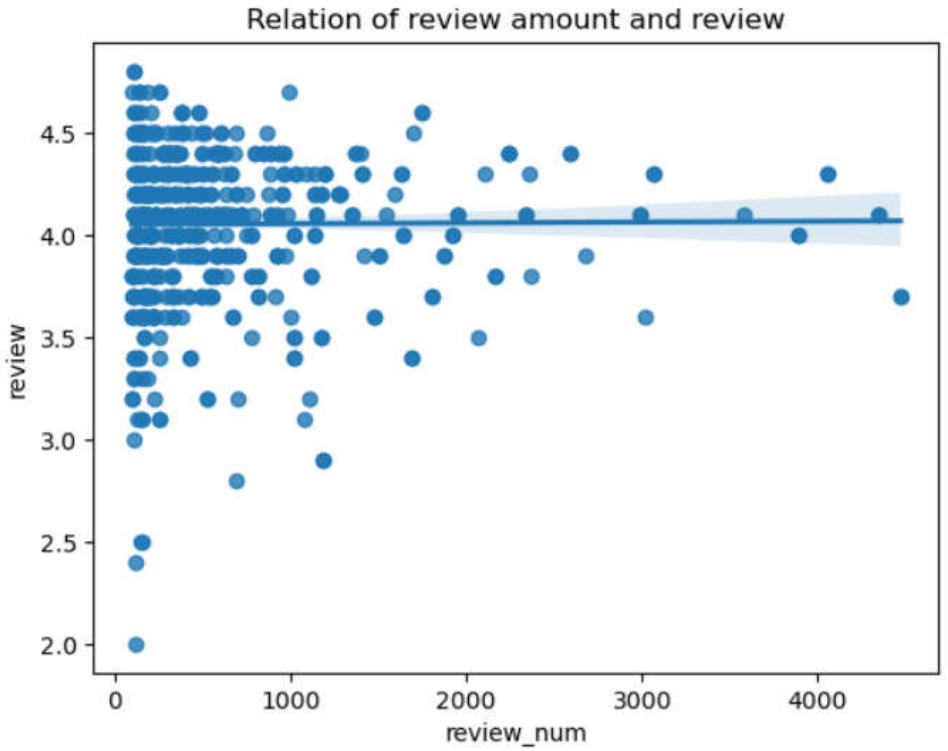
lulu[(lulu['review'] >= 4.5) & (lulu['review_num'] >= 100)]



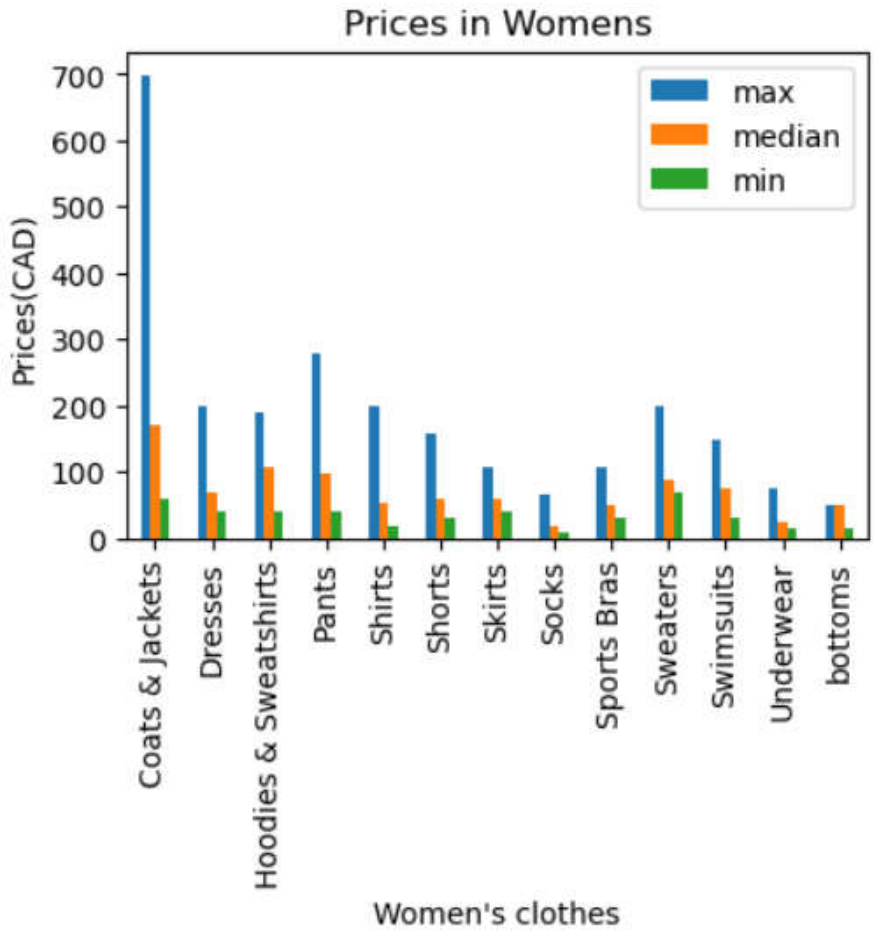


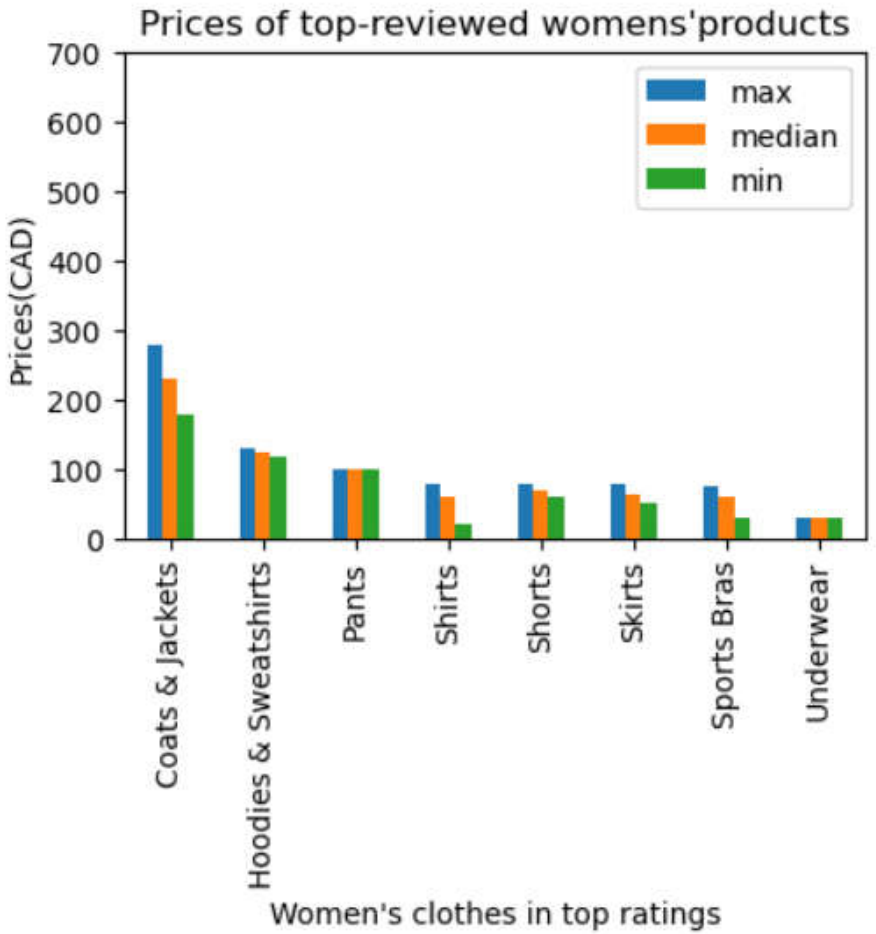
Relation of price, rating and review amount





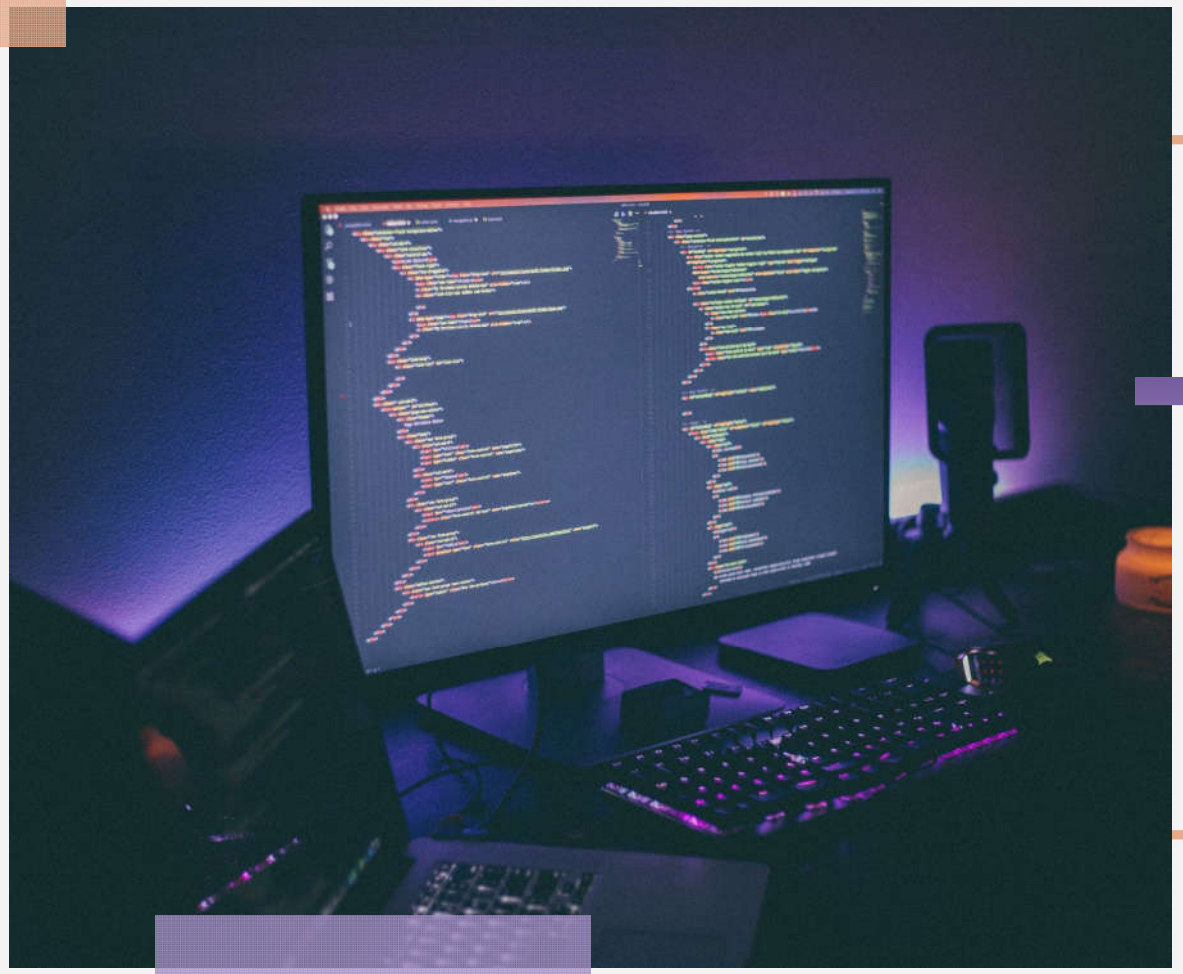
Price distribution of all products and top reviewed products





6. Next step

- Analysis of a period of price change, especially during holiday sales
- Find the detailed comments for products which obtained low review
- The relations of product texture and price
- More data required such as stock changes, revenue etc. to support further analysis



Thanks