

Detecting Price Discrimination and Price Steering on the Internet

Dat Hong, Lufan Li

Department of Computer Science, University of Iowa, Iowa City, USA

Email: dat-hong@uiowa.edu, lufan-li@uiowa.edu

Abstract—Although being relatively new topics, Internet price inconsistency including price discrimination and price steering have started attracting the attention of research community recently, especially the ones driven by personal information. Internet price inconsistency can happen due to many factors such as inventory update, distributed infrastructure or regional tax differences. In this project, we try to reproduce price discrimination based on personalization or user browser profiles. In addition, our contribution in this project is to conduct a new experiment method to detect more observable price steering. Compared to the previous work, our method can find more observable evidence of price steering on Google, and reproduce successfully price discrimination on some popular travel agent websites. Our findings reinforce the notion that user behaviour can have significant effects on price steering and price discrimination, which are increasingly taking place in today's Internet.

Keywords—price discrimination; e-commerce; tracking; price steering; personalization.

I. INTRODUCTION

Today, people are more comfortable with purchasing goods and services on e-commerce websites such as Amazon.com (most comprehensive retailer), Netflix.com (streaming TV and movies service), and Priceline.com (airline tickets and hotel stays). By purchasing on e-commerce websites, people can get many advantages. For example, these websites can personalize different customer's purchasing habits and improving user's experience. Or the online retailer can recommend similar products to the users or informing them the recent discount on the products in their wish-list or shopping-cart.

However, many websites began to take advantage of user information for one controversial purpose: manipulate the products and prices shown to the users. Recent study found that this personalization technology can induce customers to spend more money in many ways [1] [2] [3] [4] [5] [6]. One way to induce customers to spend more money called "price-steering": instead of listing the most relative products that customer is searching, these websites will put some more expensive products on the top of the list without customer's notice. The other way is known as "price discrimination": the websites may offer different prices for different users even for the exactly same product. For example, researchers have found that when different users tried to book the same room on the same day on some travel agency websites, the offered prices can

have difference up to 20% [3]. Figure 1, Figure 2 and Figure 5 are examples of price steering and price discrimination.

Detecting price or search discrimination online is not trivial. First, we need to decide which information are relevant and can cause or trigger discrimination, if it exists. There are many factors can lead to price discrimination and price steering. These factors can belong to either server side (such as inventory update, distributed infrastructure) or client-side (e.g. user profiles, operating system or location). For the purpose of this project, we exclude server-side factors. Regarding client-side factors, some recent studies have uncovered that these websites are potentially using user browser profile information such as cookies, operating system, location, and browser version [3]. For example, the travel website Orbitz was found to be personalizing the results of hotel searches [6]. Users were unaware that Orbitz "steered" Mac OS X users towards more expensive hotels by placing them at higher ranks in search results. However, based on our best knowledge, there have not been many research on price steering and discrimination based on user browser behavior.

Due to limited scope, this project is trying to detect the price discrimination and price steering only based on user behaviour such as browser profiles. More specifically, we try to build two different browser profiles, one profile for *affluent* or *elastic* customers, and one profile for *budget conscious* or *inelastic* customers. We then use these profiles to access some products in some particular categories on some popular retailer websites such as Google, Amazon, Walmart; and some popular travel agents websites such as Priceline, Cheapticket. We use Selenium to automatically do the training process and price querying process. After each training session, we save the temporary profile for later user. Before another training session or querying the prices, we use Selenium to load the temporary profile we have saved. Our results show that price discrimination and price steering can be observed even users are using the same operating system, browser, IP address, etc.

II. RELATED WORK

The problem of increasing price discrimination on the Internet was first put forward by Andrew Odlyzko [7] in 2003. However, there were not a lot of work on this until 2012, when some researchers in Universitat Politècnica de Catalunya, Telefonica Research have conducted an intensive series of research on this problem. Their first paper [1] tried to detect evidence of price discrimination and steering on many popular websites based on many different user attributes such as location, operating system, profiles. After that, they published another paper which focused on location-based price discrimination [3]. In their work, they had many server

distributed among many different places around the world. They then built a tool called Sheriff which would coordinate the requests from these servers and send to their elected website. The returned prices were collected and examined. The most recent extensive research on price steering and discrimination was from Northeastern University[3] in 2014. They monitored real users requests and developed a new approach to detect price steering, based on set theory (Jaccard Index). Their work further confirm price steering and price discrimination on the Internet.

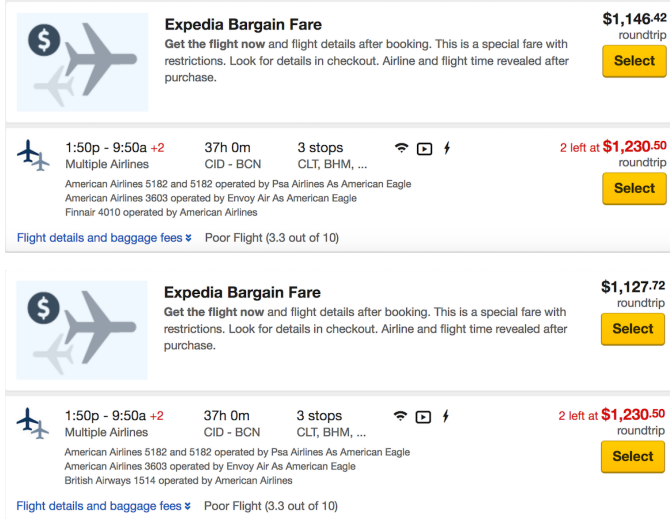


Figure 1. A price discrimination detected on Expedia.com when the query flight ticket round trip tickets. (Departing from CID on January 21st, 2017, returning from BCN on February 3rd, 2017) First result on the top is from price sensitive profile, and the second result on the bottom is from price non-sensitive profile.

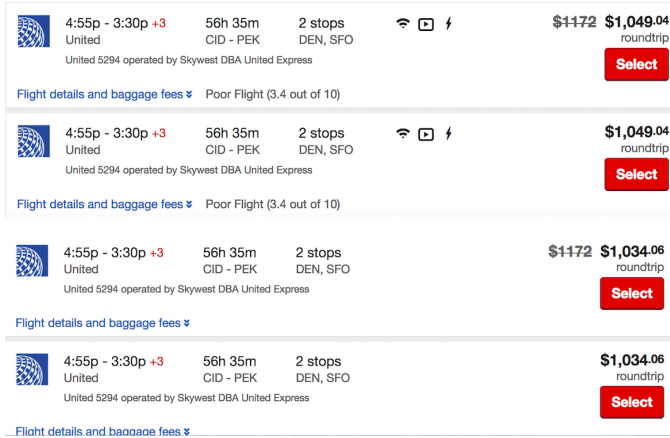


Figure 2. A price discrimination detected on Cheaptickets.com when query flight ticket round trip tickets. (Departing from CID on January 21st, 2017, returning from Beijing on February 3rd, 2017) First result on the top is from price sensitive profile, and the second result on the bottom is from price non-sensitive profile.

III. PROPOSED APPROACH

Basically, our approach to detect the price discrimination and steering is using different browser profiles to query the products prices and see if there is any price differences. The browser profile is simply a set of user data which include

user's browser preferences, browser log, searching history, cookies. The trackers (e.g. Google Analytics, Yahoo Analytics and DoubleClick) on the Internet have many ways to track users and the most common way is using cookies [6]. To build a browser profile, we need to enable browser cookie, tracker and visit some particular set of websites during a long enough period of time[1]. Previous studies showed that affluent browser profile can encounter higher prices compared to budget conscious one[1]. Our first approach is to use 2 browsers to build or train these 2 browser profiles or personas, before using them to check the price inconsistencies on some popular websites.[1] The most important part of this approach is we try to convince trackers that our profiles are really from a price sensitive user or a price non-sensitive user.

A. Prepare Links for Two User Profiles

Our hypothesis is that affluent consumers is more likely to visit luxury goods/accessories/jewelries, and some particular product segments such as estates, golf clubs, private jets, etc. On the other hand, budget conscious or price sensitive group tends to visit cheap or on-sale products, coupons, discounts, etc. Based on this, we build two blank Firefox browser profiles (one for price sensitive group and the other one for price non-sensitive group). For the first profile, which called price sensitive profile, we prepared fifty price comparison websites from Alexa.com, thirty most popular coupon websites from wistero.com, and twenty department store websites from alexa.com. For the second profile, which called price non-sensitive profile, we prepare top websites from Alexa.com and Google in many categories. All of this categories are about luxury products or services, such as jewellery, cars, personal aircraft, estates, etc.

B. Eliminating Other Factors

When detecting price discrimination and price steering, previous studies did not eliminate other factors, such as location, operating system, login status, etc [1] [3]. In our study, we will eliminate these factors and only focus on user browser profile. One reason is that if we use these factors in our study, for example, location, receiving different prices for two profiles might be reasonable, because different states in the United States might have different tax policy. The second reason is that it will be more clear for us to see how browser profile or behaviour can influent user price if we eliminate other possible factors.

In order to detect the price discrimination caused only by browser profile, we build profiles and query prices in the same computer. So, the IP address, browser version, and any other things are always same.

C. Training the browser profiles

Next, for each Firefox blank profile, we enable all trackings, cookies, and use Selenium to instruct the browsers to visit these websites. One browser we train to visit the affluent website list and another one to visit the budget conscious list. We shuffle the list of websites every training time. For each website, we randomly click on some random elements every 5-15 seconds. After clicking randomly 5 - 10

elements, we'll move on to another website. We train the browsers whenever we use our computers, and save the profiles after finishing. Our purpose is to make sure our profiles are as similar to real users as possible. In this way, we hopefully try to convince the trackers on the Internet that the user profiles we build are price sensitive user and price non-sensitive user. The training process lasted for 7 days.

D. Detecting price steering and price discrimination

To detect price steering, we begin to search several products we selected on several e-commerce websites everyday and record every product's price for each profile. According the study had done by [1], the products we selected are considered easier to detect the price difference, such as headphone, digital camera, and DVD player.

To detect price discrimination, we do the query on travel agent websites. It is easy to detect the price difference. However, this is tricky because we do not know whether our query activities affect the products' price. Products' price like flight tickets might be very fluctuating. In order to solve this problem, we query the same product's multiple times until we get the "stable" prices in latest two queries. Then, if the price for two profile are still different, we will record the prices and consider that price discrimination happened.

When detecting price discrimination and price steering, we query both flight ticket and travel bundle on three trips, from Cedar Rapid to Beijing, from Cedar Rapid to San Francisco, and from Cedar Rapid to Barcelona. All the trips' schedule is departing on Jan. 21st, 2017 and returning Feb. 3rd, 2017. We used Selenium to automatically collect the prices.

E. Websites for Querying Prices

We query products and get results from two types of websites. The first kind of website are whole sale website such as Google, Amazon, Shopping, Walmart. The reason we choose Google as the first priority is because it is known as the top tracker on the Internet. (See Figure 3).

The other type of websites is travel agent websites such as priceline.com, orbitz.com, cheapticket.com and expedia.com. We all began to query from November 29th to December 4th.

F. Modified approach for price steering

Our previous approach is inspired by [1]. However, after a week of training, we did not see any noticeable differences in the price. Even in their work, the differences are not significant[1]. To improve the results, we changed how we collect the websites for affluent browser profile. During the previous training process, we had observed that some of the products in our lists of websites appear in the advertisements and especially, our Google search results. So our next phase of

training for affluent profile, we focus only on luxury fashion websites such as Manolo Blahnik, Prada, Gucci. The training process is still the same, but we changed the keywords. We used fashion related keywords such as diamond rings, women handbags, sunglasses, wrist watch, etc. With this new approach, we can see noticeable results after only few days.

IV. RESULTS AND DISCUSSION



Figure 3. Top trackers (third parties) on top 500 Alexa websites. We can see that Google is the top tracker.

A. Price Steering on popular retail websites:

We did not observe any price discrimination from any of these websites. This is the same result as the previous work. However, we saw very clear price steering for Google search results: We examined search results for 6 of our selected keywords at the same time on November 29th. We searched several times for each query until the number of returned research stable (did not change). For most of the queries, there were significant differences in the prices of these products for different personas (Figure 4). The min, max and average prices for affluent profile were higher than budget profile and normal profile. The primary reason is because there were some exceptionally expensive products appear in the search results. These products came from the luxury fashion brand websites that we used for the training process. For instance in figure 4, the min and general prices of all 3 personals for the keyword "wrist watch" are very similar, but for the trained affluent profile, the max were usually different: there is the watch "Master Ultra Thin Moon" with price \$22900 appear in the search results. This watch comes from the website <http://www.jomashop.com/>, which is in our luxury list. This watch's price boost the whole mean and max price of the affluent profile's search results. The same phenomenon happened on other search keywords. This search results are shown in Figure 5.

Furthermore, the price steering happened even after a period of time e.g. several days. In figure 6, we can see that for keyword "diamond rings", the prices for affluent profile are generally higher than for budget conscious profile. We choose diamond rings because this type of products are already expensive. The price differences are expected not too much so we can see the differences easily.

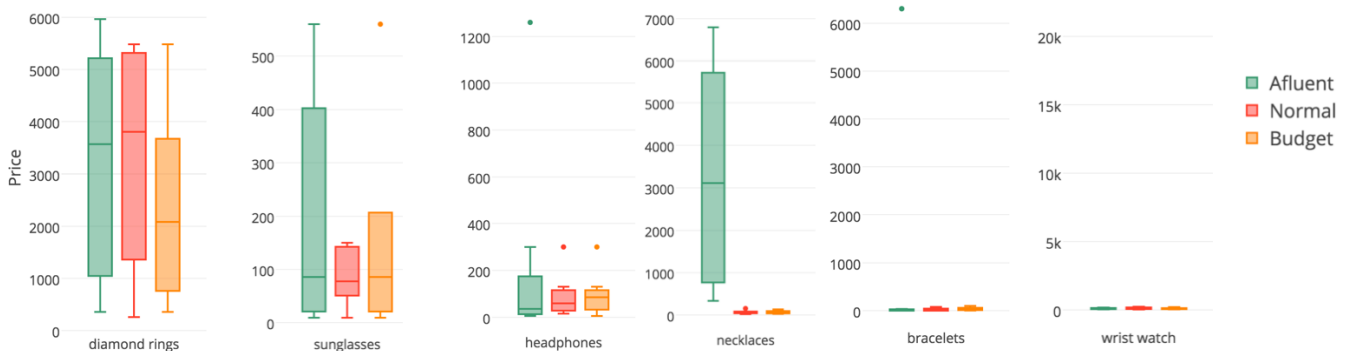


Figure 4. Boxlot of prices (min/1st percentile/mean/2nd percentile/min/max) shown by Google to the difference personas.

Not only price and search results, the advertisements , which provided by for example Google Ads, shown on some websites were also different for these personas. We observed that the products in these ads have different prices between affluent profile and budget profile. However, we do not consider this as price steering so we did not investigate it in this project. But it is very interesting to look at it in the future work, or other projects.

Finally, we checked the same queries on Amazon, Walmart, and Staples but we did not see price steering on these websites. We did see different ad contents but again, it is out of this project's scope.

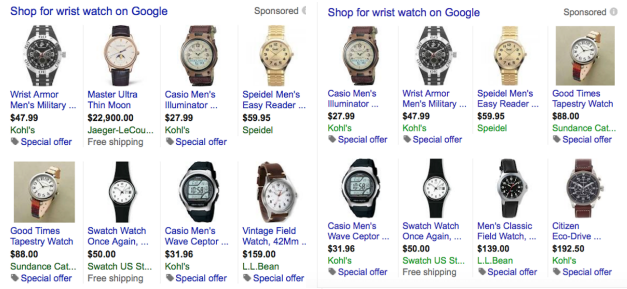


Figure 5. Price steering for “Wrist watch”. The watch “Master Ultra Thin Moon” have too high price compared to others. The left image is from affluent profile and the right is from price-sensitive one.

Summary: We did not observe price discrimination on any of these retail websites. However, we see very clear price steering, that is, the higher price products appeared in the search results for affluent personas. The reason is that for affluent profile, some products from luxury websites which were used to train the browser, appeared in the search results.

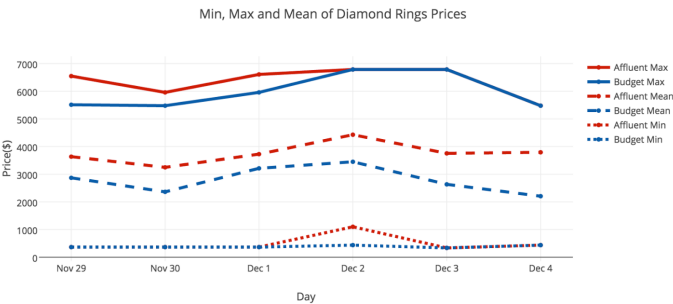


Figure 6. Recorded prices of “diamond rings” after a week from November 29th to December 4th for different personas.

B. Price Discrimination on Travel Agent Websites

There are three types of price discrimination we had found during our study. (1) Two users get different prices for the same flight ticket. When we query ticket price using both price sensitive profile and price non-sensitive profile, we might get different prices. This situation is considered as a price discrimination. Fig. 1 is an example of this type of price discrimination. (2) The tickets, which the websites recommended have the same price, but the other tickets' prices are different. In Fig. 2, we can see on the top of each result

list, the tickets' prices are the same, but the other one are different. (3) Same travel bundle have different prices. If we query the price for a travel bundle, the price might be different. See Fig. 7.

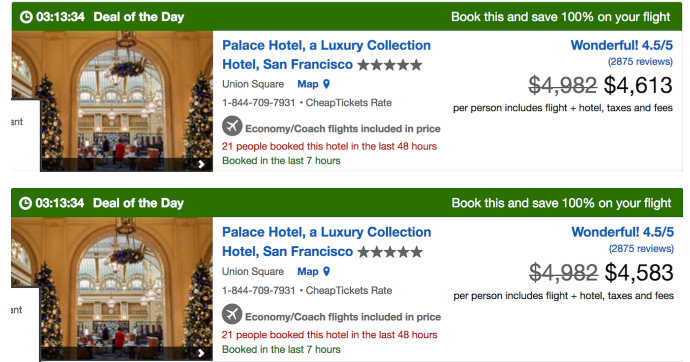


Figure 7. A price discrimination detected on Cheaptickets.com when query flight ticket round trip tickets. (Departing from CID on January 21st, 2017, returning from San Francisco on February 3rd, 2017) First result on the top is from price sensitive profile, and the second result on the bottom is from price non-sensitive profile. Also, on Cheaptickets.com, sometimes, price non-sensitive profile might have lower price.

C. Price Steering on Travel Agent Websites

We detect price steering only when we query the travel bundle's price. When we query the travel bundle's price, the travel agent websites will return us a result table. We assume that on the top of the table is the travel bundle that the websites recommended.

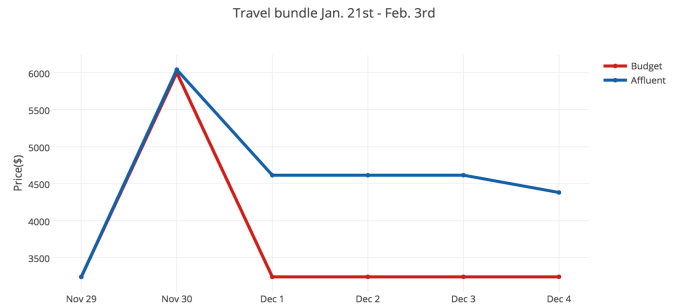


Figure 8. Price steering result on Expedia.com. (Departing from CID on January 21st, 2017, returning from San Francisco on February 3rd, 2017)

D. Expedia.com

The result on Expedia.com is much more clear than other three websites. We only found first type of price discrimination happened when we queried the round trip tickets between Cedar Rapids and Barcelona, and the price sensitive profile had the lower price than price non-sensitive profile. Third type of price discrimination happened six times on trip Cedar Rapids – San Francisco, and Cedar Rapids – Barcelona. Also, the result of price steering followed what we thought, price sensitive profile always had the lower price than non-sensitive profile or had the same price as the non-sensitive profile. See Fig. 8.

E. Priceline.com

We have not detected any price discrimination and price steering on Priceline.com. However, we found something interesting. When we query flight tickets on Priceline.com using our two profiles, we could see price difference between two profiles at the first time. However, if we refresh the page, the price difference will disappear. This situation happened a lot on Priceline.com. Although we did not mark this as price discrimination in our experiment, the high probability of first time price difference happened only on Priceline.com.

F. Orbitz.com and Cheapticket.com

Although we found both price discrimination type 1 and type 3 on Orbitz.com and Cheapticket.com, the result is not so clear to us. The price sensitive profile did not always have the lower price. The same thing happened when we detected the price steering on these two website. We do not know the reason. Our guess is that the training process was not able to convince these two websites. The other guess is that they make price discrimination happen randomly.

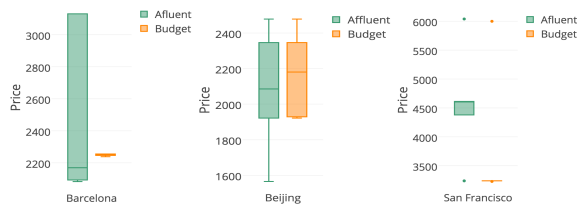


Figure 9a). Price Steering box plot of Expedia.com results returned to affluent and budget profile. (All results are the trips of departing January 21st, 2017, returning on February 3rd, 2017)

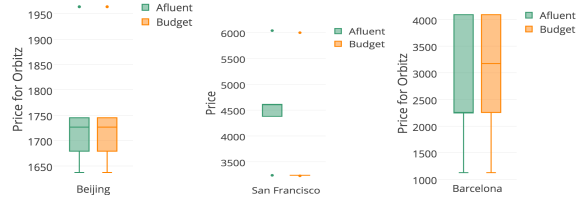


Figure 9b). Price Steering box plot of Orbitz.com results returned to affluent and budget profile. (All results are the trips of departing January 21st, 2017, returning on February 3rd, 2017)

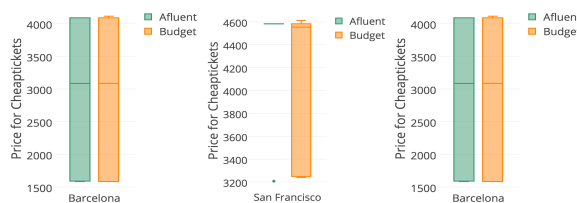


Figure 9c). Price Steering box plot of Cheaptickets.com results returned to affluent and budget profile. (All results are the trips of departing January 21st, 2017, returning on February 3rd, 2017)

V. CONCLUSION

Personalization has become a important feature of many websites and service providers in recent years. However, there is soaring evidence that e-commerce sites are using complex

personalization algorithms to implement price steering and discrimination.

Compared to previous work, in this project, we propose another browser training method which can help us detect more noticeable price steering. In our method, we focus on a particular product category e.g. fashion. Using the luxury fashion websites for training the affluent browser profile, we can observe that some of their products appear in the ads and search results. This is the primary cause of price steering in our experiment. Our measurements suggest that both price and search discrimination might be taking place in today's Internet.

Regarding price steering for travel agent websites, Although price discrimination and price steering happened a lot during our experiment, we did not find any price discrimination and steering on the trip between Cedar Rapids and Beijing. So, we conclude that price discrimination and price steering might happen only in some specific trips.

Since we know discrimination could happen on travel agent websites, we can use different methods to maintain many browser profiles on our computer even though we do not know what exactly the algorithms these websites used. In this way, we can at least get some different prices and save money.

VI. CONTRIBUTIONS

Dat Hong: Collect list of websites for affluent users, investigate price steering on popular e-commerce websites, write report about steering, write Selenium scripts for training Firefox browser profiles, write Selenium scripts to collect search results for price steering, conduct training for both profiles, collect results (both screenshots and data) for products price steering, write Python scripts to draw the graphs. Edit the report.

Lufan: Collect list of websites of budget conscious users, investigate price steering and discrimination on travel agent websites, write Selenium scripts for training profiles, write Selenium scripts to collect the data from travel websites, conduct training for both profiles, collect results (both screenshots and data) for travel agent price steering and discrimination. Write and edit the report.

REFERENCES

1. Mikians, Jakub, László Gyarmati, Vijay Erramilli, and Nikolaos Laoutaris. "Detecting Price and Search Discrimination on the Internet." Proceedings of the 11th ACM Workshop on Hot Topics in Networks - HotNets-XI (2012): n. pag. Web
2. Jakub Mikians , László Gyarmati , Vijay Erramilli , Nikolaos Laoutaris, Crowd-assisted search for price discrimination in e-commerce: first results, Proceedings of the ninth ACM conference on Emerging networking experiments and technologies, December 09-12, 2013, Santa Barbara, California, USA
3. Aniko Hannak , Gary Soeller , David Lazer , Alan Mislove , Christo Wilson, Measuring Price Discrimination and Steering on E-commerce Web Sites, Proceedings of the 2014 Conference on Internet Measurement Conference, November 05-07, 2014, Vancouver, BC, CanadaK. Elissa, "Title of paper if known," unpublished.
4. "Privacy, Economics, and Price Discrimination on the Internet." by Andrew Odlyzko :: SSRN. N.p., n.d. Web. 13 Dec. 2016.

5. Sheriff: <http://sheriff.dynu.com/views/home>
6. D. Mattioli. On Orbitz, Mac Users Steered to Pricier Hotels. The Wall Street Journal, 2012. <http://on.wsj.com/LwTnPH>.
7. Andrew Odlyzko, Privacy, economics, and price discrimination on the Internet, Proceedings of the 5th international conference on Electronic commerce, p.355-366, September 30-October 03, 2003, Pittsburgh, Pennsylvania