WEBSCRAPING LAPTOP PRICES TO ESTIMATE HEDONIC MODELS AND EXTENSIONS TO OTHER PREDICTIVE METHODS

A PREPRINT

Jean-Denis ZAFAR
Department of Consumer Prices and Household Surveys
French Institute of statistics and economic analysis
jean-denis.zafar@insee.fr

Stéphanie HIMPENS
SSP Lab Unit
French Institute of statistics and economic analysis
stephanie.himpens@insee.fr

May 6, 2019

ABSTRACT

When an item is missing and has to be replaced, the difference in quality between the dispparearing product and the new one must be taken into account in the consumer price index, in order to measure comparable prices. Hedonic regressions can be used to estimate this difference, using product characteristics as explanatory variables for the price. However, the quality of the models can be unsufficient due to the small size of samples. This paper explores the use of webscraping in order to gather bigger volumes of information on prices and characteristics, in particular for electronic goods. Traditional hedonic regressions will be compared with other predictive methods, including machine learning algorithms in terms of predictive power.

***K*eywords** consumer price index · hedonic regression · quality adjustment · webscraping

1 Introduction and presentation of hedonic prices in the consumer price index

The replacement of products used for computing price indices is a well-known problem. As CPIs are built as a measure of pure price evolutions, with fixed quality, following products along the year has been the standard methodology. It implies that products which are not sold anymore have to be replaced, to avoid attrition and selection effect. Turnover can be more or less important depending on the type of products and point of sale, but it is always necessary to take into account this phenomenon, by making sure that products are replaced properly.

The new product comes at a new price but also with new features, and the difference in prices with the replaced product can come from a quality effect as well as from a price effect (inflation). For example, a new laptop can be 5% more expensive than the previous one, but include more memory or a better processor, and then we could consider that the quality is 10% higher, leading to a negative inflation. This quality effect can be assimilated with the difference in the prices of both products at the same period. To measure it, one should have access to the price of the new product at the same time, which is practically difficult when dealing with newly sold products, or manually collected data.

If we consider that the new product is extremely similar to the previous one, the direct comparison of price - *i.e.* neglecting the quality effect - can be used. This can be the case for standardised items, or when changes are only due to fashion and products can really compare from one year to the other. However, this approach is difficult to generalise, and several methods exist to estimate the quality effect (*quality adjustment*). Implicit methods include [1]:

- Link-to-show-no-price-change (or automatic linking, or price change taken as quality change) means that the value of the quality change is assessed as the change in price since the preceding period. This method is usually avoided, as it does not take into account the inflation effect since the preceding month.
- The bridged overlap method, which is very popular: approximating the price effect between the current and the previous month, with the general evolution of a set of products having the same characteristics (e.g. same

[1] *Cf.* Eurostat manual.

consumption segment) and available during both months. Hence, the quality effect is the ratio of prices between the new and the old product, divided by the estimated evolution: $\frac{p_m^B}{p_{m-1}^A}$;

- Monthly chaining and replenishment: the aggregate relative price change between any two adjacent periods is assessed as the aggregate relative price change for the set of all product offers that are available in both those periods.

- Backcasting (base price imputation) means using the set of products which have not been replaced since the base month, to estimate the relative price change of the replacing product since that month.

As the price of a product decreases some time before it is removed from sale, many of these methods can lead to a downwards bias when linking a new product to the replaced one. This is due to the fact that the estimation of the quality effect can be overestimated, when we compare the initial price of the new product with the last price of the old one.

Conversely, other critics point out the fact that the quality of the new product can be much greater than the difference of price, and is often underestimated (see [6], [7]).

Hence, some methods use the characteristics of the products to better estimate the difference in quality, and then adjust for quality. Option pricing assumes that some part of the price comes from options (typically for cars), and then extracting the "pure" price effect between two periods can be done by removing this part of the price first. In practice, the convention is to consider that half of the option price is quality effect. Similarly, package-size adjustment assumes that the price depends on the size of the product.

Hedonic repricing generalises this concept by assuming that the quality adjustment can be done through a regression on the technical characteristics. In the context of a CPI where the current price is divided by the basis month price, this leads us to estimate the basis month price of the new product with its technical characteristics as explaining variables, enabling us to compare prices at this month for the old and the new products ([1]). Noting P the price of the product, and z_1, z_2, ..., z_k its relevant technical characteristics, the hedonic repricing model can be expressed as the following linear regression equation:

$$ln(P) = b_0 + b_1 z_1 + ... + b_k z_k + \epsilon \tag{1}$$

The difference of quality between product A and the replacing product B will then be expressed by the coefficient: $exp(b_1.(z_1^B - z_1^A) + ... + b_k.(z_k^B - z_k^A)) = \frac{\hat{P}^B}{\hat{P}^A}$, where the hat sign denotes the fitted value from the regression.

Meanwhile, more and more purchases have been made on the Internet. This has lead several countries to collect prices on the Internet, manually or through the use of webscraping, either to diminish costs of price collection or to be more representative of the household consumption. Webscraping has several applications. It can be used as a price collection for direct use in the CPI, replacing traditional manual price collection. Even though this can lead to important cuts in collection costs and better quality of data, due to the numerous observations it can lead to, it is not necessarily the easiest application of webscraping. Indeed, good infrastructure and maintenance organisation are needed to be able to collect prices for production purposes. Websites can change, or can decide to block the webscraping robots, even if they have been notified that these data are used only for public statistical purposes. As CPIs have to be computed on a monthly basis (and even twice a month, with a flash estimate and a definitive index), production constraints are particularly strong.
Automatic collection of data through webscraping also enables to gather enough data to estimate hedonic models on the basis month. It can permit us to be more representative of the different companies selling the type of product we want to study, and to save the prices of all laptops in all of the big e-commerce websites. However, on the other hand, these prices will not necesarily be representative of the ones used by physical stores (there can be offers specific to the website or to certain stores for example). Webscraping also allows us to be more exhaustive in the collection of the different technical characteristics, and then be more accurate in the choice of the features which will then be used to estimate the models. As dynamic pricing is increasingly employed, we can also imagine to use webscraping as a way to collect prices all along the month, to reach more temporal representativity.

Hedonic models are used in many contexts, from the price of housing to the price of technological goods. Here, we will focus on hedonic repricing, in the context of the consumer price index in France, and more specifically for electronic or household goods. Laptops will be the first targeted product.

2 What are the benefits of such models in the case of innovative products?

2.1 The case of innovative products

When measuring a consumer price index, it can be crucial to take into account the apparition of new products, without waiting many months. There are indeed several phenomenons that play a role in the price evolution in the innovative sectors:

- **The apparition of a new type of products**, where the problem is related to the concept of homogeneous products and consumption segments: do we create a new segment when a new product comes to the market, and then only compare the future prices of this product with the initial price of the same product to integrate this evolution in the CPI? Or do we try to compare the price of the new product with other products which are similar but less innovative and where already on the market, through the use of a bigger (less restraining) consumption segment? Typical exemples of such cases include the apparition of CD and DVD players, smartphones, Uber, etc. Integrating the new product into a category which is too loose leads to a possible underestimation of the quality increase (and then an upwards bias on the price index).
  Conversely, waiting for the next year - and basket building - before including the new product as a new segment in the basket can lead to another type of bias, if the way the new product is priced as a function of its quality and utility is different from the traditional products. In the standard case, innovations arrive at a higher cost, which would mean a downwards bias with this method. However, recently, digital innovations have sometimes led to decreases in prices or other ways of imagining payment: streaming has disrupted the payment of music and videos and come at a lower price (even though competition can enable statisticians to measure the decrease of price through its impact on DVDs and Blu-ray discs).
  In any case, hedonic models are difficult to apply to this situation.
- **The apparition of a new technical characteristic on an existing product**, for example when cameras have appeared on phones. This case is less brutal than the previous one, even if new products could sometimes be considered as traditional products to which we add one or many new characteristics. It is also difficult to handle this case with a hedonic model if the new feature is completely new and has never been observed before. However, we can try to isolate the supplement of price due to the new characteristic, and reprice the product by adding this supplement to the repricing we would have done without this new characteristic. The only problem is that we do not take into account the inflation of price of this new feature, which can be high as it can incorporate the whole novelty effect.
- **The improvement of already existing technical characteristics**. The difference in price is not necessarily the same as the impact on the value of the characteristic. As long as their prediction power is sufficient, hedonic models can be a useful tool to measure the real impact of the increase of the technical features of the product on its price. As an important part of innovation is incremental, the turnover can be high, while the main characteristics tend not to differ within a year. We will see that this is particularly the case with electronic goods, for example when a new processor incorporates more cores, or higher frequency, or when the laptop has more RAM.
- **The impact of the new products on the price of older products**. Through competition, new products entering the markets with new characteristics have an impact on the prices of the products already on the market. For example, when a phone with new features (*e.g.* smartphones) appeared, the prices of standard phones decreased. For similar quality, one can pay a lower price because the traditional product is obsolete and attracts fewer consumers. Hence, this part of the impact of innovation can be measured through standard price collection, while the product is still sold.

2.2 How we plan to use webscraping for hedonic models in France

The specificity of our approach is that, even if we have working webscraping robots, we do not want to use them on a regular basis yet. We want our infrastructure to be fully robust before using it in production. This involves maintaining our bots efficiently, monitoring data and coordinating webscraping tools with standard CPI applications. Among the recent developments, using scanner data in production has been the priority in France. We then first plan to use manually collected prices throughout the year, combined with replacements with hedonic repricing based on data webscraped at the basis month. Hedonic models have not been used widely in the past, due to the cost of collecting detailed information on the technical characteristics of the products, and to the insufficient number of observations to estimate models. We want webscraping to provide us with more information on the products. The basket will be keeped fixed.

Given the advantages of webscraping to collect prices of more products at more frequent times, it is possible that it will be gradually used in production in the next few years. In this case, the use of hedonic models with time dummy

variables (where the inflation estimator between times 0 and m directly comes from the regression) will be an option, but it is important to be very cautious with the estimation, as we would have to perform the regression analysis every month. It can generate more work for the price statisticians, and a lack of precision if the analysis is not done properly. Overlap methods could also be efficient in this context, as we might capture the price of the replacing product and the old one simultaneously. Nevertheless, they can lead to a bias if the quality adjustment coefficient is not computed at periods of the life-cycles of both products which make the prices comparable ([2]).

Hence, we will only consider in the paper the use of hedonic models for repricing the substitute product at the basis month. Even if it also necessary to lead a consistent statistical analysis for this purpose, the variable cost is limited as the models are to be reestimated only every year.

2.3 Why keeping the same adjustment coefficient?

We could wonder whether such models are useful for repricing, as we might think that choosing the substitute product in the database collected at the basis month would guarantee that we can get the actual observed price, and then do not need any statistical model. This would be a mistake, and even for products which were already present at the basis month, we rather want to use the repricing model. Taking the actual observed price can lead to a bias: if some products are more "permanent" than other ones, we want to be representative of the whole set of products that we started to follow at the basis month. An "observed" repricing would completely eliminate the choice of the previous product in the sample.
More formally, let p_i, t be the logarithm of the price $P_{i,t}$ of a product i at month t, X_i its technical characteristics. The hedonic models writes:

$$p_{i,t} = X_i * \beta + \epsilon_{t,i} \tag{2}$$

with β the estimated regression coefficient, $\epsilon_{t,i}$ the residual and $\mathbb{E}_{j \in products_{month0}}(\epsilon_{j,t=0}) = 0$. The residual term adds the estimation error of the model, and an inflation term.

We will note $f(X_k) = X_k * \beta$ (f can also be any function linking the technical characteristics of the product to its price). We can split the set of products present at the basis month into two categories, P the ones that will continue to be present until month t (the most "permanent"), and the ones that will disappear D. We have :

$$\mathop{\mathbb{E}}_{j \in P}(\epsilon_{j,t=0}) + \mathop{\mathbb{E}}_{j \in D}(\epsilon_{j,t=0}) = 0 \tag{3}$$

For types of products with strong innovation, P and D are likely to behave differently, and so are the respective expectation terms. When we chose a replacing product at time t, we want to be fully representative of the set of products which are present at t. Conversely, when we reprice the product for month 0 (basis month), we want it to fully reflect the set of products at time 0. With a fixed basket approach, it means that we want the repricing of the replaced product to be consistent of the structure of the set of product at time 0. More specifically, if we consider the following choices for (log-)repricing product k_0, which replaces product j:

- if $i_0 \in P$, taking the actual price $p_{k,0}$ at month 0
- $f(X_k)$, that is using the raw hedonic model estimation (fitted value)
- $p_j + f(X_k) - f(X_j)$, *i.e.* adding the price due to the differences of technical characteristics between j and k to the price of j at basis month

The first choice (using directly the price of the product collected in our webscraping database if it was already present) has a bias, as its associated ϵ represents set P, where we want to reflect D because $j \in D$. The second estimation is better, because it is neutral ; with a perfect model, it would be the closest approximation of a representative price of a product with characteristics X_k at month 0. However, we will even prefer the third one, because it is more representative of products like j, but reajusted to estimate a fictional j which would have had technical characteristics X_k; j has been drawn in the sample, so we want to continue to use it even if we replace it with k and reestimate the basis month price accordingly. Finally, the quality adjustment coefficient will be $\frac{p_{k,t}}{p_{j,t=0}+f(X_k)-f(X_j)}$

Furthermore, the use of this coefficient avoids us to make strong assumptions on the price behaviours online and in physical stores. We do not suppose that the expectation of a computer with a given set of technical characteristics is the same in both types of store; we only need the hypothesis that the expected difference in prices between two products is the same on the website and in the outlet.

3 How and what do we webscrape?

For our purpose, webscraping programs have been developed, using Python. Prices from three French e-commerce websites have been collected: rueducommerce.com, boulanger.com and darty.com. These websites are among the most famous and most used in France. We only worked on the two first ones for the purpose of this study. For each website, the result pages have been scraped for the chosen products, which permits us to collect the price (this is the price the consumer sees when looking for a product, and almost always the same price as the transaction price without delivery costs), the name and brand of the product, and the URL of the detailed product page. We also have the possible discount rate, which will enable us to use either the actual price with discount of the initial price before discount. The detailed product page is then itself scraped, in order to gather information about the technical characteristics of the product. For such websites, the concept of consumption segment is quite close to the categories visible on the site architecture. They have thus been chosen to determine the URL of the research page, instead of looking for a particular keyword. For example, the laptops have their own categories on each website.

Webscraping has intrinsic difficulties. First, the website can change, which can lead to a disability of the program initially written and a need to recode it. Maintenance is thus very important in production. For hedonic models purposes this matter is less difficult to tackle, as the robot has to be executed only when a reestimation of the coefficients is needed, i.e. at each base month (December) for hedonic repricing. The programs can also be blocked by the websites. In order to avoid this, it is important to be sure that the websites are aware of the nature of the price collection that National Statistical Institutes (NSIs) are performing: this is a statistical work of general interest, published only at aggregated level, with strong commitment to keep the data confidential. The possible legal arguments (compulsory surveys, etc.) should also be taken into account when contacting them. However, this problem is a greater concern when dealing with webscraping in production, as we are then more likely to being blocked.

The databases contain the name of the product, the brand, the price and all of the technical characteristics visible on the website. This leads to a large number of variables. Moreover, these databases, despite the fact that they bring overall better quality than manual collection due to their systematic aspect and the quantity of data which is generated, also have drawbacks, decreasing the quality:

- some of these variables are underfilled for an important proportion of the observations ;
- variables named differently may correspond to the same concepts ;
- modalities can be different for identical or similar products

Hence, a lot of cleaning and variable selection work has been done. Infrequent brands and seldom used colour terms have been grouped together to limit sparse modalities. Missing values have been imputed with the most frequent value (qualitative variable) or the mean value (continuous variable) of filled observations. Imputed values should be stable between two estimates of the hedonic model at the risk of introducing an additional quality effect. Removing all observations with missing values would have led to the deletion of too many observations (only observations with more than 50% of the missing variables were deleted). In future works, other missing values imputation ($e.g.$ hotdecks) can be performed to improve the quality of imputation.

3.1 Data from rueducommerce.com

Rueducommerce.com is one of the main online retailers in France. It proposes more than 3 millions of products. Originally focused on electronics products, it nowadays also sells household, gardening products as well as clothes. In 2016, it was bought by Carrefour, which is one of the biggest traditional retailers in France. There are more than 12000 Carrefour stores in the world, including around 250 hypermarkets and 1050 supermarkets in France [2].

The *rueducommerce.com* website was scraped twice in 2019 : the first time on the 26th of February and the second time on the 25th of March (less than one month after the 26th of February). Data scraped in February contains 380 observations (we selected only laptops distributed by Rueducommerce seller or by a local store[3] and that had more than 50% of the scraped variables not missing. We also dropped products that were sold by Wewoo and Yonis, as these too brands were not selling laptops on the platform.

From all the features available on the website, we built and selected 34 variables. Many were categorical variables (such as brand, colours...). Using too many variables in an hedonic model is not possible, as there would be a risk of overfitting. It would lead to poor results in prediction. Hence, we had to select only a subset of these variables.

[2] http://www.carrefour.com/fr/content/les-supermarches

[3] Laptops that were referred as only sold in local store and not on the platform

On rueducommerce.com, several sellers are present (in particular discount type sellers). In order to avoid any effects related to the seller, we limited ourselves to laptops sold by Rueducommerce and in local stores by Carrefour. Refurbished or used computers were also excluded from the analysis.

Table 1: Number of observations after cleaning steps

	Scraped data on February	Scraped data on March
Initial number of scraped laptops	1109	1153
Number after deletions of products sold by wevoo and yonis	990	1032
Number after deletions of second hand laptops	911	933
Number after deletions of observations with more than 50 % of missing values	805	787
Number after removal of sellers other than rueducommerce and carrefour	380	377

Among the 380 laptops present in February, 323 were still present one month later in March. 57 have disappeared and 54 have appeared between February and March.

3.2 Data from boulanger.com

The website boulanger.com has also been scraped. Boulanger is a well-known retailer specialised in household and electronic goods. It has been bought by the big retail company Auchan. There are 130 Boulanger stores in France.
Laptop prices have been scraped on this website on 25th January and 25th March. For January, we get 359 prices of laptops, and 421 for March. More precisely :

- 263 were present in January and March
- 120 had the same price in January and March
- 95 have disappeared
- 158 have appeared, most of them being probably new

Data coming from boulanger.com are cleaner than the observations coming from rueducommerce.com, because this website is not a platform like rueducommerce.com. However, an important work has also been done to choose variables that can be used, in order not to remove too many observations.

3.3 Complementary data

Data coming from the websites may be insufficient, which has lead us to complete them with other characteristics. For the processors, many observations were missing, while the brand and model were correctly filled. We have then scraped Intel's and AMD's websites, in order to gather more accurate information on the base and boost frequency, the cache, the number of cores, the date of launch, which was used in complement and possibly substitution of the data from the websites.
The price of the processor is also an information available on the websites, but we have not kept it for possible use as an explaining variable. This is not a fixed characteristic of the processor, as its price evolves between the basis month and the moment when the hedonic model is used. However, we could think of using the price in a different way from the other characteristics, by keeping its value in the basis month to estimate the price of the laptop at that month. This implies that the processor already existed, which is not necessary the case. Hedonic models are to be used for new computers, which are likely to contain innovative characteristics, among which a new processor is a good candidate.

4 Automatic selection of a subset of the explaining variables

The data we get from the e-commerce websites are quite exhaustive to describe the products. Many of these characteristics can contain redundant information, for example the length and width of the laptop and the size of the screen, or the base and turbo frequency, etc. Moreover, there are far too many variables because:

- we do not want to overfit our model
- many of the variables may have small importance
- in practice, if manual collection is done on a monthly basis, we do not want the collection to cause too much work for the price collectors.

A model with all these variables would not be possible in practice.

Hence, it is important to select a subset of our set of variables. To do this, we have mainly used two types of methods (see [3]):

- **Methods based on trees**. In regression trees, the observations are divided into homogeneous price groups according to the values of their characteristics. In these methods, the predicted value is the average of the prices in each group. The variable selection method consists in cutting the tree at a given level and retrieving all the variables (or nodes) that make it up.
- **Shrinkage methods**. They consist in performing a regression with an extra term that penalises the value of the coefficients. Ridge or LASSO regressions make the coefficients of the least predictive variables decrease, or even cancel out. It is then sufficient to observe the parameter values in order to know the most relevant features of the scraped database.

Here, we first use these methods as subset selection algorithms for future use of hedonic models. LASSO are generally mainly used for this purpose, but it can also lead to predictions (fitted values with the computed coefficients). This is even more the case for regression trees, which are one of the most commonly used prediction methods in machine learning. We will then use these algorithms both for subset selection and for price prediction.

The methods are estimated each month on a training sample of 80% of the scrapped data from the rueducommerce.com and boulanger.com. The accuracy of the results obtained is then assessed on the 20% of the remaining data. To do this ensures that the model previously estimated is not over-fitting the data.

For each of the methods, we estimate the accuracy of the predicted values mainly by two type of statistics, using respectively L1 and L2 norms:

- The Mean Absolute Error : $\sum_{i=1} \frac{1}{n}|y_i - \hat{y_i}|$ and its variant, the accuracy defined as : $\frac{1}{n}\sum_{i=1}^{n}(1 - \frac{|y_i - \hat{y_i}|}{y_i})$
- The Mean Squared Error : $\sum_{i=1} \frac{1}{n}(y_i - \hat{y_i})^2$, which penalises more values for which there is a big prediction error.

where y_i are the true prices and $\hat{y_i}$ are the predicted values.

Throughout this document we tested two types of models to predict prices :

- We predicted the prices in levels : $y_i = \sum \alpha_i x_i$ (predictors and features are unchanged)
- We predicted the logarithm of prices : $log(y_i) = \sum_i x_i$

4.1 A tree-based method: random forest

Tree based methods separate observations according to a tree. Each node splits observations in two branches according to the values of a specific explanatory variable. The most relevant features (the ones which cause the highest decreases in the intra-classes variances) are at the top of the tree. They make it possible to separate very heterogeneous data sets and thus significantly reduce variance within two levels of the tree. It is thus possible to extract a model by cutting the tree and selecting corresponding nodes. By cutting the tree and keeping the upper part, we obtain a selection of the most explanatory variables of the prices. It is also possible to identify some breaks in the predictive power of some continuous variables. Trees are essentially non-linear predictors. Tree-based methods have some weaknesses: building trees is performed randomly. It depends on the sample and sometimes on some initialisation parameters. Random forests is a method allowing some of these boundaries to be bypassed. It consists in estimating a large number of trees on samples of the initial data. If the sample is made without replacement (*i.e.* meaning possible duplicates in the sample), it is called bagging, otherwise it is called boosting. The general idea is that combining several estimators of learning methods gives better results than using a single estimator. It enables us to limit the volatility of the results obtained with a single tree by combining several results. The algorithm does not only use a sample of the initial data to build each tree. It also uses a sample of the variables of the database. This leads to more biased and dispersed samples for each of the elementary trees but to a better final estimator by combining all the results.

4.1.1 Results

We estimated random forest on data collected in February and March on rueducommerce.com, and January and March for boulanger.com . As the final aim of the models is to predict a price for a given month, we have chosen not to mix data from different months. We tried to predict the logarithm of prices as well as the untransformed prices. The accuracy of both models, for both websites, can be found below. Even if the algorithm can be fitted on transformed prices, we have chosen to compute the prediction metrics on the untransformed prices, in order to be able to compare the performance.

The results are highly dependent on the training and test samples used. On a random draw of 100 samples, mean accuracies of the models were ranging from 78% (untransformed prices on initial prices) to 87% (logarithm initial prices) in January and February. The results on log and untransformed prices are quite similar.

Table 2: Accuracy of random forests (100 samples)

	Rueducommerce.com				**Boulanger.com**			
	February		**March**		**January**		**March**	
	Discounted prices	**Initial prices**	**Discounted prices**	**Initial prices**	**Discounted prices**	**Initial prices**	**Discounted prices**	**Initial prices**
Accuracy - no logarithm	85%	86%	85%	85%	85%	86%	83%	83%
Accuracy - logarithm	86%	87%	86%	86%	86%	87%	82%	83%
MAE - no logarithm	182	194	173	195	161	145	217	218
MAE - logarithm	179	199	196	200	155	142	253	256
MSE - no logarithm	72299	94589	70634	96180	52930	39945	122404	130617
MSE - logarithm	72011	106688	103247	109283	39724	31332	216831	246838

The results obtained using all dummies for graphics cards were performing better, but it is less reliable in real life as it is more dependent on the models currently sold. If a new graphics card appears, the model would not take it into account. However, it is also possible to split the set of graphics card available in the market into big categories of similar cards and create dummies for each category.

Accuracy is always close to 85%, which is rather bad for a machine learning model. Predicting the logarithm of the prices or the discounted prices does not seem to change anything to the weak results of the model. The probable reason is that some major characteristics of a laptop computer are very simplified in the model: for example, there are multiple graphics cards, and only the brand and type are involved in the modelling; RAM is almost a qualitative variable but it is considered as a continuous variable.
However, to select the variables for the use of the hedonic model, such prediction errors are not necessarily a problem. To do this, we retain the most important variables, i.e. those that lead to the greatest variance reduction in the nodes of the trees where they appear.
Even for repricing in a hedonic approach, such errors do not invalidate the model. They mean that even for a given set of technical characteristics, we can get a high variance of prices.

For each random draw, we selected features whose importance is greater than 0; we have used scikit-learn package in Python. Price modelling is largely based on RAM (from 63% to 79% of reduction in variances on average, this feature is always selected). We discarded some features that only appeared in very few samples (the brand of the graphics card and the screen size). We selected the presence of backlit keyboard. However this variable is important in March but not in January nor February.

The precision is better with random forests than for simple based-tree models. However, they lose in interpretability what they gain in precision. It is no longer possible to know the precise combination of variables used to draw a single tree.It is still possible to get an idea of the combination of variables used to build the trees. Below, we draw the 3 highest levels of two trees chosen randomly.

In both trees, RAM intervenes at the top. It separates the sample into two groups with respect to 7. This is consistent with the importance calculated by the scikit-learn package. The tree methods are essentially non-linear. The RAM variable is continuous. The link with the price could be non-linear.

In March, the use of variable selection of the models on logarithm prices in linear regressions leads to very poor results. In the data set, a few observations can have a strong impact on the results. For example, on rueducommerce.com, one computer disrupts the estimate when it is in the test sample. This computer shows an exceptionally high value for its SSD capacity (2000 Go). It is the only item to have such a value. The effect of SSD capacity might be non linear. In the next chapters we choose to use the logarithm of features to predict the logarithm of prices in linear models. The model

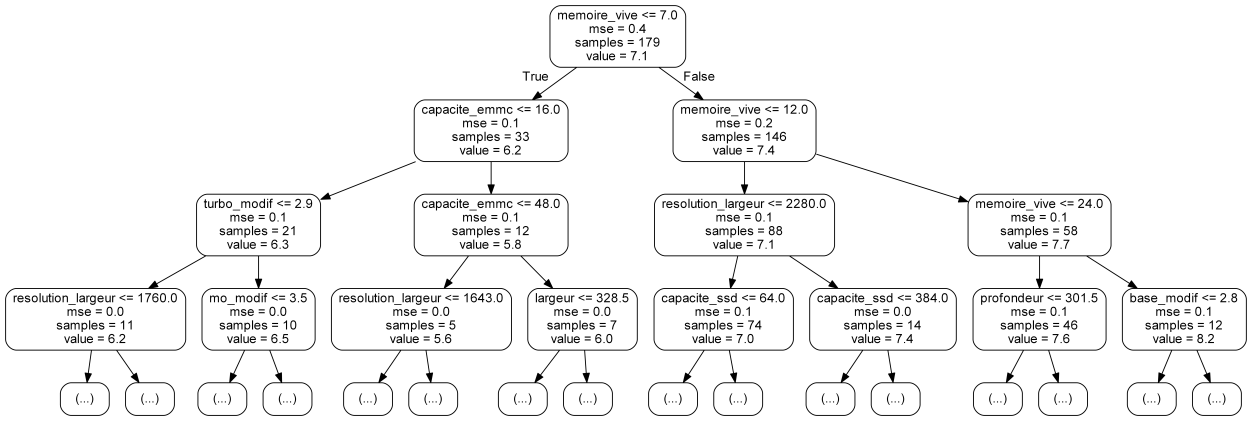
Table 3: Selected features (100 samples) - untransformed prices

	Rueducommerce.com				Boulanger.com			
	February		March		February		March	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
processor_base_frequency	100	100	100	100	100	100	100	100
ssd_capacity	100	100	100	100	100	100	100	100
graphics_card_type_gtx	95	80	91	95	1	24	48	100
graphics_card_type_rtx	100	87	100	100	0	0	100	100
graphics_card_brand_amd	51	0	2	0	0	0	100	0
backlit_keyboard	0	0	8	14	0	0	81	99
processor_number_of_cores	61	96	94	97	0	0	0	0
colours_gray	6	0	0	6	0	0	0	0
colours_metal	1	0	0	0	0	0	0	0
colours_black	3	0	0	0	0	0	0	0
height	100	100	100	100	100	100	100	100
width	98	100	100	100	100	100	100	100
brand_apple	25	1	43	7	59	0	100	0
brand_asus	3	1	0	1	0	0	0	0
RAM	100	100	100	100	100	100	100	100
processor_cache_size	100	100	100	100	100	100	100	100
weight	100	100	100	100	100	100	100	100
depth	100	100	100	100	100	100	100	100
screen_resolution_height	44	99	56	83	100	100	100	100
screen_resolution_width	95	100	97	100	100	100	100	100
processor_boost_frequency	100	100	100	100	0	0	0	0
screen_size	0	3	0	0	11	100	0	0

Table 4: Selected features (100 samples) - logarithm of prices

	Rueducommerce.com				Boulanger.com			
	February		March		February		March	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
processor_boost_frequency	100	100	100	100	100	100	100	100
processor_cache_size	100	100	100	100	100	100	100	100
processor_base_frequency	100	100	100	100	100	100	100	100
weight	100	100	100	100	100	100	100	100
height	100	100	100	100	100	100	100	100
depth	100	100	100	100	100	100	100	100
RAM	100	100	100	100	100	100	100	100
ssd_capacity	100	100	100	100	100	100	100	100
operating_system_mojave	100	25	92	71	0	0	0	0
screen_resolution_width	98	100	100	100	100	100	100	100
width	80	98	95	100	100	100	100	100
screen_resolution_height	77	100	80	80	100	100	100	100
graphics_card_type_gtx	75	25	63	68	0	0	0	100
brand_other	65	14	35	7	0	0	0	0
graphics_card_type_rtx	54	9	90	68	0	0	0	0
emmc_size	48	54	85	33	-	-	-	-
processor_number_of_cores	33	78	80	95	0	0	0	0
brand_apple	25	2	23	6	0	0	100	0
graphics_card_brand_nvidia	11	3	1	0	0	0	0	1
screen_size	3	5	5	6	98	100	11	90
graphics_card_brand_amd	2	0	0	0	0	0	0	0
backlit_keyboard	0	2	33	43	0	0	78	100

Figure 1: One of the trees generated by a random forest (prediction of the logarithm of initial prices in March



becomes :

$$log(y_i) = f((log(x_i)_{i\ Quantitative}, (x_j)_{j\ Qualitative})$$

4.2 A shrinkage method : LASSO regression (least absolute shrinkage and selection operator)

We want to estimate a linear model. It would therefore be advisable to use a linear method to determine the best subset of the initial variables to be retained. It is possible to take a step-by-step approach by introducing (or removing) the characteristics successively. However, this can be very costly in terms of calculation time. It is not suited to big amount of features and data.

Another set of methods consists in introducing all the explanatory variables into the model and placing constraints on the value of their coefficients. The Ridge and Lasso methods try to reduce the value of the coefficients. This has been proved to reduce the variance of the estimates. These methods are based on linear regression. The difference here is that they add an additional constraint to penalize for high coefficient values . For Lasso, the penalization is the sum of the absolute values of the coefficients, and the amount to minimise becomes :

$$\underset{\alpha_1,\ldots,\alpha_p}{min} \frac{1}{2}\sum_{i=1}^{n}(y_i - \alpha_0 - \sum_{j=1}^{p}\alpha_j x_{i,j})^2 + \lambda\sum_{j=1}^{p}|\alpha_j|$$

The constraint aims at shrinking the coefficients as in the Ridge regression. Unlike this later method, the introduction of L1 norms allows some of the coefficients to be cancelled strictly, where the L2 regression used in Ridge does not remove them. The LASSO regression can thus be used to select a subset of features.

The lambda parameter is used to control the intensity of the regulation.The higher it is set, the more parameters of the model are turned to zero and discarded. This parameter can be selected by cross-validation.

4.2.1 Results obtained on the rueducommerce database

As we did with random forests, we estimated a LASSO regression to predict prices and then the logarithm of prices. The lambda parameter is selected by cross-validation. We split the database in five folds. The model was estimated on the four folds and validated on the remaining fold. The number of observations is not very large. The results are quite volatile. Firstly, we compared the accuracy of the models on discounted and initial prices. The accuracies and mean squared errors are reported below.

The accuracy of the model on untransformed prices slightly declines in March. The results of this model are less stable than those of the model estimated on logarithm of prices. They highly depend on the parameter λ selected by cross-validation. To check the robustness of the results we randomly selected 100 training samples. The average accuracy are not very far from what we previously got.

Table 5: Accuracy of LASSO models (100 samples)

	Rueducommerce.com				Boulanger.com			
	February		March		February		March	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
Accuracy - no logarithm	83%	81%	84%	81%	79%	79%	78%	78%
Accuracy - logarithm	85%	85%	85%	85%	83%	82%	83%	82%
MAE - no logarithm	200	237	189	241	199	201	207	227
MAE – logarithm	201	232	197	236	215	230	215	249
MSE - no logarithm	297	360	281	361	275	277	296	320
MSE - logarithm	301	368	308	383	408	414	439	506

To select the best subset of variables, we choose the variables with the highest coefficients in the LASSO regression. Finally the estimated coefficients are :

The selected subset of variables is broadly stable over the two months. There are some slight differences between the set of features selected by the models.

Compared to random forest, screen resolution is only selected to predict the logarithm of prices. The coefficient of the screen resolution height is also very small. The dimensions of the laptops (width, height and depth) do not matter. Other features such as the presence of a backlit keyboard, some of the colours or of a certain type of graphics card (rtx) are also included in the model.

We also tried to include a indicator featuring memory size larger or equal to 7.0 (like in the random forests). This improves slightly the accuracy of the model (to reach 85%). This tends to prove that the effect of the memory size on prices is not linear. It is also possible to introduce the squared value of the memory size in the models. It also seems to improve the predictive power of our models (however the effect is still moderate).

4.2.2 Results obtained on the combination of both databases

The databases are rather small: they only contains a little more than 300 observations. We stacked observations coming from both websites to get a more robust model. In production, it would also imply an easier estimation, as it would avoid having several different models (this would also avoid multiple reestimations every year for each type of product). We try to fit a linear regression on a small subset of 16 variables simultaneously present in both databases. We first modelled untransformed prices, which lead to disappointing results. The coefficients are highly volatile on the three databases. The accuracies are low.

The accuracy of the linear model on all data is around 81%. A lasso regression discard four variables : seller (rue du commerce or boulanger), weight, backlit keyboard and screen size. The accuracy of the linear regression drop to 78%.

The coefficients are sometimes very different from one database to the other. The variables may be filled differently. The pricing policies could be different between two websites. The number of variables is quite low. Some key features might have been omitted. The model seems to be very much influenced by high prices. It could be a good idea to remove some highly influencial laptops.

Only some of the characteristics are common to both bases. Some laptops may be present on both website. It would be possible to compare their collected features.

Table 6: Selected features (lasso on logarithm of prices)

	Rueducommerce.com				**Boulanger.com**			
	February		March		February		March	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
processor_base_frequency	100	100	100	100	80	97	93	100
Hard_disk_capacity	83	95	22	82	59	59	100	93
emmc_capacity	90	83	46	32	0	0	0	0
ssd_capacity	100	100	100	100	100	100	100	100
graphics_card_type_gtx	33	57	16	56	60	42	79	75
graphics_card_type_rtx	100	100	100	100	1	0	100	99
graphics_card_brand_amd	100	97	100	98	0	0	2	2
graphics_card_brand_nvidia	68	71	12	31	70	80	100	99
graphics_card_ti	15	46	8	26	0	0	0	0
backlit_keyboard	97	100	100	100	100	100	100	100
processor_number_of_cores	71	69	49	66	100	100	98	98
colours_blue	85	83	26	63	40	38	75	32
colours_gray	69	56	40	43	24	31	9	55
colours_metal	66	79	98	96	3	0	2	0
colours_black	86	94	86	94	37	68	32	22
colours_gold	46	54	81	50	4	24	29	21
touch_screen	100	100	100	100	89	92	79	94
height	80	76	33	62	100	100	100	100
width	12	35	3	18	0	0	0	0
DVD_player	35	80	37	56	0	0	0	0
brand_acer	98	95	89	96	1	5	0	0
brand_apple	100	100	100	100	100	100	100	100
brand_asus	50	57	21	49	0	0	17	32
brand_dell	64	84	35	56	0	0	0	0
brand_hp	47	77	51	43	0	2	11	5
brand_lenovo	87	97	91	97	2	3	0	0
brand_microsoft	98	98	81	100	12	13	7	15
brand_msi	85	70	100	79	62	60	89	63
RAM	100	100	100	100	100	100	100	100
processor_cache_size	100	100	100	100	100	100	100	100
weight	100	100	100	100	99	99	91	65
depth	73	95	44	94	0	0	0	0
screen_resolution_height	99	96	100	74	5	8	3	2
screen_resolution_width	10	31	2	54	98	98	100	100
operating_system_chrome	92	89	57	78	15	13	9	3
operating_system_mojave	100	98	97	96	39	37	82	72
operating_system_none	100	100	100	100	0	0	0	0
screen_size	32	44	40	37	87	89	92	48
processor_boost_frequency	100	100	100	100	100	100	100	99
local_store_only	79	97	67	91	0	0	0	0
battery_life	0	0	0	0	100	100	98	100

Note: each cell represents the number of samples for which the corresponding feature is selected.

Table 7: Selected features (100 samples) - lasso on untransformed prices

	Rueducommerce.com				**Boulanger.com**			
	February		March		February		March	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
processor_base_frequency	100	100	100	100	90	100	83	100
Hard_disk_capacity	86	99	89	98	8	4	91	56
emmc_capacity	96	92	99	98	0	0	0	0
ssd_capacity	100	100	100	100	100	100	100	100
graphics_card_type_gtx	77	87	88	64	24	31	7	61
graphics_card_type_rtx	100	100	100	100	0	0	100	100
graphics_card_brand_amd	71	67	42	25	5	8	80	40
graphics_card_brand_nvidia	20	60	36	51	0	0	0	1
graphics_card_ti	94	42	70	34	0	0	0	0
backlit_keyboard	100	100	100	100	5	18	1	8
processor_number_of_cores	100	100	91	95	99	99	99	97
colours_blue	65	43	39	29	0	0	3	2
colours_gray	36	61	51	49	0	0	1	21
colours_metal	41	55	58	31	82	56	2	0
colours_black	97	100	100	100	0	2	16	17
colours_gold	35	53	50	33	0	0	0	0
touch_screen	99	100	100	100	0	5	8	12
height	52	43	19	25	100	100	100	100
width	68	40	76	30	0	0	0	0
DVD_player	49	73	51	48	0	0	0	0
brand_acer	61	51	73	53	0	1	0	0
brand_apple	100	100	100	100	100	100	100	100
brand_asus	49	60	40	64	0	0	2	59
brand_dell	84	97	61	65	0	0	0	0
brand_hp	43	96	53	74	1	0	5	0
brand_lenovo	77	94	77	88	3	2	1	1
brand_microsoft	99	98	99	100	0	0	1	10
brand_msi	93	87	100	92	0	3	4	0
RAM	100	100	100	100	100	100	100	100
processor_cache_size	100	100	100	100	100	100	100	100
weight	100	100	100	100	100	100	94	81
depth	20	53	6	50	1	8	23	27
screen_resolution_height	100	99	99	76	0	0	6	0
screen_resolution_width	0	24	6	18	100	100	100	100
operating_system_chrome	32	43	21	28	0	1	3	2
operating_system_mojave	87	78	29	69	0	0	4	0
operating_system_none	100	100	100	100	0	0	0	0
screen_size	100	100	100	100	22	21	9	2
processor_boost_frequency	97	87	97	98	74	54	85	1
local_store_only	86	98	58	94	0	0	0	0
battery_life	-	-	-	-	96	94	99	100

Note: each cell represents the number of samples for which the corresponding feature is selected.

Table 8: Accuracy of linear regression

	All		Rue du commerce		Boulanger	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
Accuracy - logarithm	81	81	83	83	83	82
MAE - logarithm	230	258	232	261	215	256
RMSE - logarithm	338	421	367	424	365	389
Accuracy - no logarithm	76	77	81	80	78	77
MAE - no logarithm	232	258	238	262	216	239
RMSE - no logarithm	313	399	347	395	315	332

Table 9: Coefficients estimated on stacked databases

	All			
	Logarithm		Untransformed	
	Discounted prices	Initial prices	Discounted prices	Initial prices
(Intercept)	2.44 ***	2.90 ***	-1571.1 ***	-1470.4 ***
emmc_capacity	-0.02 *	-0.03 *	2.03 **	1.81 *
ssd_capacity	0.04 ***	0.05 ***	0.97 ***	1.05 ***
backlit_keyboard	0.04 .	0.025	26.42	-12.98
processor_number_of_cores	0.13 ***	0.10 **	69.26 ***	49.36 *
touch_screen	0.11 ***	0.15 ***	73.27	124.80 *
brand_apple	0.40 ***	0.41 ***	642.6 ***	663.2 ***
brand_microsoft	0.17 **	0.24 ***	340.03 ***	492.5 ***
RAM	0.585 ***	0.57 ***	60.26 ***	70.91 ***
processor_cache_size	0.20 ***	0.19 ***	57.73 ***	59.6 ***
weight	0.04	0.06 *	0.004	0.36
screen_resolution_width	0.21 ***	0.19 ***	0.09 .	0.05
operating_system_mojave	0.17 **	0.11 *	232.38 **	164.7 .
screen_size	0.31 *	0.26 .	75.47 ***	80.73 ***
processor_turbo_frequency	0.10 *	0.09 *	9.29	2.80
graphics_card_none	-0.06 *	-0.12 ***	-10.98	-89.92 .
seller_rueducommerce	-0.05	-0.05	-77.71	-57.43

Note: for each coefficient, the significance of the p-value is represented: 0 < '***' < 0.001 < "**"< 0.01 <"*" < 0.05 < "." < 0.1 < " " < 1.

5 Accuracy of the hedonic models

In this part we estimated linear models retaining the subsets of data previously selected. The best model will be the one able to predict observed prices the most accurately. We also used these models to predict a change in price.

5.1 Predicting prices with the subsets of variables selected by random forests

We introduced all of the variables selected in one of the random forests in a linear regressions to estimate price changes. We tried to predict the transformed and untransformed prices. To take the logarithm of prices has one main advantage : it always produces positive estimates. A few estimates were negative with the regression in levels. We had to drop them from the calculation.

The accuracy of the linear models is lower than the one of the random forests. Random forests are non linear estimators. They are fitted to the complex characteristics set of laptops. But they can't select the best subset of features to use in linear regression. The average accuracy on the 100 randomly drawn samples is lower. It is below 86 %. We used a subset obtained by non linear method to estimate a linear model. That's why there is such a drop in accuracy.

We then want to use apply our models to inflation estimation. We estimated the price evolutions on boulanger.com data, between January and March, using the hedonic model with the random forest variable selection (January plays the role of the basis month). Each item present in January was selected in the fixed basket. For these products, if the price was missing in March, a hedonic replacement was made, with a basis month repricing. The substitute product was chosen

Table 10: Coefficients estimated separately on the two databases (common features)

	Rue du commerce				Boulanger			
	Logarithm		Untransformed		Logarithm		Untransformed	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
(Intercept)	6.36 ***	6.49 ***	-31.4	538.7	1.62*	1.98 *	-1580.1 ***	-1413.2 ***
emmc_capacity	-0.03 *	-0.03	0.815	1.12	0.014	-0.004	2.73 **	2.30 *
ssd_capacity	0.04 ***	0.05 ***	1.06 ***	1.20 ***	0.04 ***	0.05 ***	1.05 ***	1.21 ***
backlit_keyboard	0.02	-0.02	96.44 *	-15.14	0.05	0.09 *	13.50	79.35
processor_number_of_cores	0.09 .	0.09 .	91.28 ***	73.72 **	0.09 .	0.05	35.13	17.75
touch_screen	0.13 **	0.16 ***	126.9	255.3 **	0.09 .	0.12 **	36.34	48.56
brand_apple	0.38 ***	0.39 ***	624.9 ***	602.13 **	0.39 ***	0.37 ***	527.37 ***	481.7 ***
brand_microsoft	0.24 **	0.29 ***	339.9 *	365.6 *	0.12 .	0.20 **	242.8 *	430.8 ***
RAM	0.57 ***	0.56 ***	36.80 ***	46.2 ***	0.56 ***	0.56 ***	82.02 ***	86.62 ***
processor_cache_size	0.22 ***	0.22 ***	59.89 ***	63.75 ***	0.13 **	0.12 **	41.15 **	43.36 **
weight	0.37 ***	0.40 ***	441.0 ***	575.0 ***	0.02	0.04	0.09	0.43
screen_resolution_width	0.037	0.03	0.07	-0.03	0.27 **	0.27 **	0.15	0.16
operating_system_mojave	0.23 *	0.15	265.1	240.4	0.14	0.08	171.0	87.7
screen_size	-0.69 *	-0.69 *	-82.1 **	-121.9 ***	0.24	0.27	60.1 **	73.0 ***
processor_turbo_frequency	0.04	0.07 .	4.36	10.66	0.73 ***	0.48 ***	72.8	-8.3
graphics_card_none	-0.07 .	-0.09 *	71.1	43.85	-0.07 .	-0.12 **	-41.3	-113.9 .

Note: for each coefficient, the significance of the p-value is represented: 0 < '***' < 0.001 < "**"< 0.01 <"*" < 0.05 < "." < 0.1 < " " < 1.

Table 11: Accuracy of regressions on subsets of features determined by random forests (100 samples)

	Rueducommerce.com				Boulanger.com			
	February		March		February		March	
	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
Accuracy - no logarithm	81 %	79 %	80 %	78 %	78 %	78 %	77 %	78 %
MAE - no logarithm	217	251	215	275	203	205	211	229
RMSE - no logarithm	299	361	307	401	277	280	292	313
Accuracy - logarithm	85 %	84 %	84 %	83 %	84 %	83 %	84 %	83 %
MAE - logarithm	212	251	210	256	195	211	189	224
RMSE - logarithm	313	400	323	418	317	325	316	341

Note: the RMSE is the root-MSE

randomly among all of the products available in March. The price aggregation for each month was computed with a Jevons index. A Dutot index would perhaps be less advisable, as the products are quite heterogeneous. This estimation was only made on the data obtained with webscraping, as the manually collected data currently contains unsufficient information on the technical characteristics. We did the same between March and April, March being the basis month. Theses estimates were then compared to the ones we would get with a bridged overlap, again using Jevons index. Sofar, the differences are quite small (the change in the source of data leads to a much bigger difference, as the index could be quite close to 100 between January and March).

Table 12: Comparison of price indices between bridged overlap and hedonic regression, Boulanger.com data

	March/January with basis month = January	April/March with basis month = March
Bridged overlap	95.8	98.3
Hedonic model	96.3	98.5

5.2 Predicting prices with the subsets of variables from the Lasso regression

We selected all of the 23 variables selected in one of the LASSO models of part 2. These features were used in four linear regressions : on levels and transformed prices on one hand and on discounted or not prices on the other hand.

The accuracies of the models are comparable to those obtained with the subset retained by random forests. However there are seven extra variables in the LASSO models compared to the previous ones. Some features such as colours and

Table 13: Coefficients of linear regression (variables selected by random forests)

	Rueducommerce.com				**Boulanger.com**			
	Logarithm		**Untransformed**		**Logarithm**		**Untransformed**	
	Discounted prices	**Initial prices**	**Discounted prices**	**Initial prices**	**Discounted prices**	**Initial prices**	**Discounted prices**	**Initial prices**
(Intercept)	2.92 ***	3.40 ***	-851.7 ***	-829.8 ***	1,39	1,22	-5246.4 **	-6175.5 ***
processor_base_frequency	0.32 ***	0.25 ***	67.89.	31.95	0,012 ***	0,176 **	-75.0	119.1
ssd_capacity	0.049 ***	0.052 ***	1.02 ***	1.2 ***	0,039 *	0,045 ***	13.47	24.62 *
graphics_card_type_gtx	-0.006	0.022	-170.1 **	-170.6 **	0,10 ***	0,13 **	159.8 *	236.1 **
graphics_card_type_rtx	0.27 ***	0.26 ***	338.9 ***	153.63 .	0,48	0,446 ***	1238.5 ***	1318.3 ***
graphics_card_brand_amd	-0.125 *	-0.106 .			0,035	0,048	289.3 **	265.6 *
graphics_card_brand_nvidia	-0.023	-0.041			0,031	0,039	-92.7	-106.3
backlit_keyboard	-0.084 **	-0.13 ***			0,012	0,032	-124.3 *	-111.8 *
processor_number_of_cores	0.035	0.017	38.2 .	19.0	0,049	0,004	147.28 .	92.62
height	-0.054 .	-0.049	-0.75	-0.21	1,36 ***	1,56 ***	1999.8 ***	2834.6 ***
width	-1.28 ***	-1.54 ***	-4.14 ***	-6.20 ***	-0,758 *	-0,57 .	-991.9 .	-1152.2 *
brand_apple	0.27 **	0.18 .	502.7 ***	353.7 *	0,40 ***	0,31 ***	597.0 ***	487.9 ***
RAM	0.49 ***	0.50 ***	28.7 ***	39.213 ***	0,46 ***	0,48 ***	651.6 ***	708.1 ***
processor_cache_size	0.209 ***	0.21 ***	68.0 ***	69.8 ***	0,16 ***	0,18 ***	136.3 *	178.9 **
weight	336.6 ***	429.6 ***			0,12	0,037	273.1 .	150.0
depth	1.31 ***	1.56 ***	5.48 **	8.54 ***	-0,28 ***	-0,27 ***	-182.9	-184.1
screen_resolution_height	0.99 *	1.71 ***	1.35 ***	2.42 ***	-0,32	-0,70 *	711.4	158.0
screen_resolution_width	-0.82 *	-1.57 ***	-0.64 **	-1.33 ***	0,66 *	1,03 ***	-103.2	487.3
operating_system_mojave	0.12	0.05	81.2	46.4	0,056	0,018	43.0	-18.5
screen_size	0.64 ***	0.71 ***			-0,27	-0,46	-1389.2 **	-1678.1 **
processor_boost_frequency	0.14 ***	0.17 ***	21.1	34.6	0,71 ***	0,40 ***	606.6 **	249.8

Note: for each coefficient, the significance of the p-value is represented: 0 < '***' < 0.001 < "**"< 0.01 <"*" < 0.05 < "." < 0.1 < " " < 1.

screen_resolution are never significant. The average accuracies of the 100 samples are slightly below. The decline is not so big since the LASSO models are linear.

Table 14: Accuracy of linear regressions on 100 samples (LASSO subset of features)

	Rueducommerce.com				Boulanger.com			
	February		March		February		March	
	Discounted prices	Initial prices	|Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
Accuracy - logarithm	86%	86%	85%	85%	78 %	78 %	77 %	77 %
Accuracy - no logarithm	83%	82%	83%	80%	78 %	78 %	77 %	77 %
MAE - no logarithm	198	237	185	237	211	213	215	233
RMSE - no logarithm	283	344	266	353	290	293	294	324
Accuracy - logarithm	86%	86%	85%	85%	82%	82%	82%	81%
MAE - logarithm	199	219	191	228	215	225	203	237
RMSE	283	331	279	349	336	342	314	363

Note: the RMSE is the root-MSE

Here, the price indices estimates are even closer to the bridged overlap ones than what we got with the random forest selection.

Table 15: Comparison of price indices between bridged overlap and hedonic regression, Boulanger.com data

	March/January with basis month = January	April/March with basis month = March
Bridged overlap	95.7	98.3
Hedonic model	96.1	98.2

Table 16: Coefficients of linear regression (variables selected by LASSO regressions)

	Rueducommerce.com				Boulanger.com			
	Logarithm		Untransformed prices		Logarithm		Untransformed prices	
	Discounted prices	Initial prices	|Discounted prices	Initial prices	Discounted prices	Initial prices	Discounted prices	Initial prices
(Intercept)	4.7 ***	4.7 ***	-41.5	241.6	3.56 ***	3.68 ***	-608.94	-684.8
processor_base_frequency	0.34 ***	0.27 ***	115.4 **	97.2 .	0.081	0.273***	6.42	134.2
emmc_capacity	2.81 *	3.12 *			0.001 *	0.001	2.9532 **	3.03 **
ssd_capacity	0.055 ***	0.058 ***	1.04 ***	1.22 ***	0.046 ***	0.049 ***	33.8 **	40.7 **
graphics_card_type_rtx	0.30 ***	0.27 ***	588.4 ***	479.67 ***	0.41 ***	0.40 ***	355.3	263.01
graphics_card_brand_amd	-0.069	-0.068			-0.022	-0.022	200.1 .	223.5 *
backlit_keyboard	-0.075 **	-0.12 ***	-96.1 *	-162.4 ***	0.012	0.033	-73.0	-63.0
processor_number_of_cores	46.7 *	41.8 .			0.10 *	0.074.	201.1 *	203.61 **
colours_black	-0.056 *	-0.061 *	-122.9 **	-152.7 **	0.018	0.025	-59.3	-56.0
touch_screen	0.16 ***	0.17 ***	173.5 **	228.2 ***	0.138 ***	0.148 ***	146.5 *	146.6 *
brand_acer	0.114 **	0.1 *			-0.048	-0.031	79.9	81.2419
brand_apple	0.42 ***	0.40 ***	856.9 ***	766.1 ***	0.55 ***	0.47 ***	733.7 ***	712.0 ***
brand_lenovo	-0.062	-0.073.			-0.039	-0.064.	-25.5	-42.4068
brand_microsoft	0.19 **	0.27 ***	297.7 **	368.0 **	0.22 ***	0.23 ***	419.9 ***	437.7 ***
brand_msi	102.4 *	84.637			-0.03	-0.12 .	165.17	173.9
RAM	0.46 ***	0.47 ***	26.9 ***	33.7 ***	0.48 ***	0.50 ***	752.4 ***	735.9 ***
processor_cache_size	0.27 ***	0.26 ***	65.3 ***	66.6 ***	0.19 ***	0.23 ***	174. **	200.3882 **
weight	0.27 ***	0.31 ***	438.3 ***	599.0 ***	0.095	0.064	332.7 **	340.7 **
operating_system_mojave	0.139 .	0.083			0.11	0.087	69.1890	48.5664
operating_system_none	-0.19 ***	-0.20 ***	-336.6 ***	-338.6 ***	'-	'-		
screen_size	-71.5 ***	-106.4 ***			0.16	0.25	-626.5	-558.9086 **
processor_boost_frequency	0.15 ***	0.18 ***	27.5	32.6	0.80 ***	0.43 ***	726.0 ***	585.1

Note: for each coefficient, the significance of the p-value is represented: 0 < '***' < 0.001 < "**"< 0.01 <"*" < 0.05 < "." < 0.1 < " " < 1.

6 Conclusion

This article shows the application of webscraping for hedonic model estimation, for electronic goods and more specifically laptops. Automatic data collection can be used at the basis month for extracting complete information on models and technical characteristics, permitting us to use machine learning algorithms to select most relevant variables. These algorithms were also used for prediction, as the goal of hedonic models is to reprice the product chosen at basis month with the technical features of the new product. The results of the models fall between 78% and 85% of accuracy on linear models, and between 83% and 87% for random forests, showing that machine learning prediction methods can be a promising way to reprice substitute products. However, the difference is not very large, and linear models would be used without losing too much precision. Moreover, the errors can be caused by a high variability of pricing by the seller, even for given technical characteristics.
Further work is still in development or planned, including:

- the simulation of different techniques on price indices over a larger period
- using random forest and LASSO predictors directly in the hedonic models to estimate the quality change
- using generalized additive models (e.g. with cubic splines)
- scraping more websites, including price comparison websites
- a better treatment of missing values, etc.

References

[1] Eurostat, *HICP manual*, 2018.

[2] International Monetary Fund, *Update of the Consumer Price Index Manual*, draft, 2019.

[3] G. James, D.Witten, T. Hastie and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*, Springer, 2013

[4] E. Diewert, S.Heravi, M. Silver, *Hedonic Imputation versus Time Dummy Hedonic Indexes*, NBER Working Paper No. 14018, 2008

[5] M. Leclair, L. Aeberhardt *et al.*, *L'économie numérique fausse-t-elle le partage volume-prix du PIB ?*

[6] P. Aghion, A. Bergeaud, T. Boppart, P. Klenow et H. Li, *Missing Growth from Creative Destruction*, NBER Working Paper, n w24023, 2017

[7] Boskin Commission (1996), *Toward a more accurate measure of the cost of living: final report*, http://www.ssa.gov/history/reports/boskinnrpt.html

[8] J. Bascher, T. Lacroix, *Dish-washers and PCs in the French CPI: hedonic modeling, from design to practice*, 1999

[9] E.L.Groshen, B.C. Moyer, A.M. Aizcorbe, R. Bradley et D.M. Friedman, *How Government Statistics Adjust for Potential Biases from Quality Change and New Goods in an Age of Digital Technologies: A View from the Trenches*, Journal of Economic Perspectives, vol. 31(2), pp. 187-2102017