STATBEL
België in cijfers

# Webscraping, de verzameling en verwerking van online data voor de consumptieprijsindex

- Ken Van Loon en Dorien Roels -

nr. 02

ANALYSE

01.2018

economie
FOD Economie, K.M.O., Middenstand en Energie
.be

# Webscraping, de verzameling en verwerking van online data voor de consumptieprijsindex

**Dorien Roels, Ken Van Loon**[1]

[1] Statistici bij Statbel (Algemene Directie Statistiek – Statistics Belgium)

## ABSTRACT

Deze analyse licht het gebruik van webscraping in de consumptieprijsindex toe. Wat houdt webscraping in? Hoe zien deze data er uit? En hoe verwerkt Statbel (AD Statistiek – Statistics Belgium) deze gegevens?

Gezien het groeiende belang van webwinkels en de opkomst van de online verkoop bij de 'klassieke winkels', acht Statbel het noodzakelijk om ook deze gegevens mee in rekening te brengen in de berekening van de consumptieprijsindex. Webscraping biedt in vergelijking met de klassieke prijsopnames bovendien het voordeel dat de prijs van een veelvoud van producten kan worden opgevolgd,

Webscraping is een techniek om automatisch gegevens van webpagina's af te halen *('scrapen')*. Statbel gebruikt de programmeertaal R om webscraping uit te voeren. De geprogrammeerde scripts worden op regelmatige tijdstippen automatisch uitgevoerd en de verzamelde gegevens worden centraal bewaard voor verdere verwerking. Een dashboard applicatie werd ontwikkeld om de uitvoering van de scripts te monitoren. Daarnaast werd een robottool gecreëerd om de prijswijzigingen voor specifieke producten op te volgen.

Statbel voerde de afgelopen maanden allerlei case studies uit. Voor specifieke producten, zoals consumentenelektronica, schoenen, hotels, tweedehandswagens en studentenkamers, kunnen de gegevens immers online verzameld worden,. Statbel vergaarde de gegevens voor de bovenstaande productgroepen en testte verschillende methoden voor indexberekening op basis van webscraping uit.

Uit de resultaten van de case studies rond consumentenelektronica en tweedehandswagens blijkt dat kwaliteitskenmerken het mogelijk maken om berekeningen uit te voeren die moeilijk realiseerbaar zijn via de klassieke berekeningsmethode. Voor de productgroep hotels biedt webscraping bijvoorbeeld de mogelijheid om de steekproef uit te breiden, hetgeen de kwaliteit van de gegevens ten goede komt. En de case studie voor schoenen toont aan dat webscraping resultaten genereert die vergelijkbaar zijn aan de klassieke prijsopnames gedaan in fysieke winkels.

Dit artikel beschrijft daarnaast ook een aantal algoritmes ter uitvoering van machine learning. Machine learning is de studie waarbij algoritmes ontwikkeld worden zodat machines/computers/programma's zelf kunnen 'leren'. Hierdoor worden niet enkel voorgeprogrammeerde acties uitgevoerd, maar kan het programma zichzelf ontwikkelen. Via patronen uit geclassificeerde data – supervised machine learning - leert het programma bijvoorbeeld nieuwe ongeclassificeerde data te klasseren. Deze classificatie gebeurt o.b.v. herkenning van woorden en codes. Verschillende algoritmes werden getest en op dit moment maakt Statbel reeds gebruik van supervised machine learning in het kader van scannerdata.

# INHOUD

*Webscraping, de verzameling en verwerking van online data voor de consumptieprijsindex* 1

*Abstract* 2

*Inhoud* 3

*1. Inleiding* 4

*2. Big data* 5

*3. Implicaties op de berekening van de CPI* 6

*4. Webscraping: hoe werkt het?* 7

*5. Case Studie – Consumentenelektronica* 11

**5.1. GEKS & RYGEKS** 12

**5.2. Hedonische regressie** 15

5.2.1. Time Dummy Hedonics 16

5.2.2. Time Dummy met Rolling Window 20

5.2.3. Fixed Effects met Window Splice 22

5.2.4. Time Dummy met Window Splice 23

**5.3. Bedenkingen bij hedonische methoden** 26

*6. Case Studie – Tweedehandswagens* 27

*7. Case Studie – Hotels* 29

*8. Case Studie – Studentenkamers* 31

*9. Case Studie – Schoenen* 33

*10. Machine Learning* 40

**10.1. KNN** 41

**10.2. Support vector machine** 41

**10.3. Naive Bayes classifier** 43

**10.4. Random Forests** 44

**10.5. Case Studie - Kledij** 45

**10.6. Case Studie – dvd's & Blu-rays** 46

*11. Dashboard* 48

*12. Robottool* 50

*13. Conclusie* 53

# 1. INLEIDING

De consumptieprijsindex (CPI) is een maandelijkse statistiek die opgemaakt wordt door Statbel (AD Statistiek - Statistics Belgium van de FOD Economie). Het is een economische indicator die de prijsevolutie meet van consumptieve bestedingen van Belgische consumenten. Het is de belangrijkste maatstaf voor inflatie. De CPI vormt in België via de gezondheidsindex en de afgevlakte index rechtstreeks de basis voor de indexering van pensioenen, sociale uitkeringen, fiscale barema's, huur en sommige lonen en wedden.

De CPI wordt berekend op basis van een korf van goederen en diensten die door de gezinnen worden aangekocht en als representatief voor hun verbruiksgewoonten kan worden beschouwd. Aangezien het aanbod van goederen en diensten continu verandert, wijzigt ook de steekproef van de opgenomen prijzen regelmatig. Momenteel worden prijzen van goederen en diensten opgevolgd voor 229 productcategorieën.

De prijsopvolging gebeurt op basis van verschillende databronnen. Zo worden prijzen ingezameld door enquêteurs die winkels bezoeken verspreid over het land. De gegevensinzameling voor de huurenquête gebeurt via papier of web. De prijzen met het grootste gewichtsaandeel worden echter centraal ingezameld via websites, catalogussen, telefoon of via bestanden verkregen van regulatoren of privébedrijven. Recentelijk worden ook meer bigdata-bronnen gebruikt in de berekening van de consumptieprijsindex, namelijk scannerdata van supermarktketens en webscraping.

Naast de nationale consumptieprijsindex (CPI) berekent Statbel ook een Europees geharmoniseerde consumentenprijsindex (Harmonized Index of Consumer Prices, HICP). De HICP maakt een vergelijking tussen de inflatiegraden van de lidstaten van de Europese Unie mogelijk. De toegepaste bestedingsoptiek en methoden zijn daartoe gecoördineerd en in Europese regelgeving vastgelegd. De uitkomsten van de CPI en de HICP zijn echter niet gelijk. Dat komt vooral door een andere weging en samenstelling van het pakket goederen en diensten waarop deze indices zijn gebaseerd.

Dit artikel geeft meer inzicht in de techniek van webscraping en de verwerking van de verzamelde data in de berekening van de consumptieprijsindex. Webscraping is een techniek om automatisch gegevens van webpagina's af te halen *('scrapen')*. Webpagina's zijn echter opgemaakt uit HTML-codes en bevatten gigantisch veel data in textvorm. Het is dus nodig om die data op een gestructureerde manier te verzamelen en te verwerken, zodat ze gebruikt kunnen worden voor statistische doeleinden. De implementatie van webscraping vergt dan ook een andere soort van vaardigheden, in het bijzonder programmeren, dataverzameling en -verwerking. In de context van consumptieprijsindex gaat de interesse uit naar het verzamelen van prijzen en productomschrijvingen op webshops.

Het groeiende belang van webwinkels en de online verkoop van 'klassieke winkels' maken het noodzakelijk om ook deze gegevens mee in rekening te brengen in de berekening van prijsindices. Het gebruik van deze data laat ook toe om de efficiëntie van de dataverzameling te verbeteren. Daarnaast verhoogt het ook de kwaliteit van de statistieken.
Dit artikel geeft weer hoe dat kan gebeuren. Het is als volgt opgebouwd:

- introductie tot de begrippen 'big data' en webscraping;
- hoe voert Statbel webscraping uit?
- prijsevolutie van consumentenelektronica en verschillende methodes voor de indexberekening van dit soort data;
- casestudies & methodes: tweedehandswagens, hotelkamers, studentenkamers en schoenen;
- machinelearning & methodes om gescrapete data automatisch te klasseren of om bepaalde observaties uit te sluiten;
- webscraping-dashboard om scripts te monitoren & robottool om prijswijzigingen voor specifieke producten te observeren;
- huidige stand van zaken.

Webscraping biedt ook voor andere statistieken een interessant alternatief voor de klassieke directe bevraging. In dit kader onderzoekt Statbel onder meer de mogelijkheid om via deze techniek het aantal vacatures te meten. Het gebruik van webscraping voor andere doeleinden dan de CPI, valt echter buiten de reikwijdte van deze analyse.

## 2. BIG DATA

Is webscraping big data? De bekomen datasets bevatten namelijk geen miljarden records. Toch kan webscraping vanuit het concept van prijsstatistiek als big data beschouwd worden Klassieke prijsopnames voor schoenen leveren ongeveer 15.000 prijzen per jaar. De website van één winkel dagelijks scrapen zorgt voor meer dan een miljoen prijzen per jaar. Die gegevens verwerken tot indexcijfers, moet op een meer geautomatiseerde manier gebeuren dan bij klassieke prijsopnames. De methoden die gebruikt worden bij de verwerking van klassieke gegevensinzameling kunnen dan ook niet toegepast worden op deze online data.

Meer data wil uiteraard nog niet zeggen dat de data beter is. De uiteindelijke doelstelling is om de online prijzen te gebruiken om de inflatie te ramen, eventueel als proxy (benadering) voor 'offline' prijzen of als aanvulling van klassieke prijzen.

Een potentieel voordeel is wel dat de frequentie van prijsinzameling opgedreven kan worden via webscraping. Aangezien webscraping dagelijks kan gebeuren, betekent dat uiteraard een verbetering. Online prijzen worden momenteel ook al via manuele gegevensinzameling als proxy genomen voor offline prijzen. De prijsinzameling voor vele consumentenelektronica gebeurt bijvoorbeeld ook al online, zonder webscraping. Prijzen worden dan centraal genoteerd via websites van handelaars. Dat is niet alleen in België het geval, maar ook in de meeste Europese landen.

Het Billion Prices Project[2] van MIT (Massachusetts Institute of Technology) toonde ook aan dat inflatie die (dagelijks) geraamd werd op basis van online prijzen zeer dicht tegen de officiële inflatie lag die berekend wordt door het Bureau of Labor Statistics.

Momenteel is dit project gecommercialiseerd via een spin-off (PriceStats[3]) en wordt de inflatie van koopkrachtpariteiten geraamd voor 22 landen (waaronder onze buurlanden). Maar: zowel in het Billion Prices Project als voor PriceStats worden de officiële inflatiecijfers gebruikt voor segmenten waar de prijzen moeilijk online kunnen gemeten worden (dit zijn voornamelijk diensten en energie). Waar precies officiële cijfers gebruikt worden en waar enkel online prijzen is niet-publieke informatie.

In het kader van de consumptieprijsindex stelt er zich geen juridisch probleem om aan webscraping te doen. Zo stelt de regelgeving voor de Europese geharmoniseerde consumptieprijsindex (HICP)[4] dat statistische eenheden indien nodig gegevens dienen te verstrekken voor de berekening van de HICP:

> *3. De statistische eenheden die gegevens verstrekken over producten die zijn opgenomen in de monetaire consumptieve bestedingen van de huishoudens, werken waar nodig samen bij het verzamelen of verstrekken van de basisgegevens. De statistische eenheden verstrekken de nationale instanties die belast zijn met de samenstelling van de geharmoniseerde indexcijfers, nauwkeurige en volledige basisgegevens.*[5]

[2] http://www.thebillionpricesproject.com/
[3] https://www.pricestats.com/
[4] Naast de nationale consumptieprijsindex (CPI) berekent Statbel ook een Europees geharmoniseerde consumentenprijsindex (Harmonized Index of Consumer Prices, HICP). De HICP maakt een vergelijking tussen de inflatiegraden van de lidstaten van de Europese Unie mogelijk. De toegepaste bestedingsoptiek en methoden zijn daartoe zo goed mogelijk gecoördineerd en in Europese regelgeving vastgelegd. De uitkomsten van de CPI en de HICP zijn niet gelijk. Dat komt vooral door een andere weging en samenstelling van het pakket goederen en diensten waarop deze indices zijn gebaseerd.
[5] Verordening (EU) 2016/792 artikel 5.3

# 3. IMPLICATIES OP DE BEREKENING VAN DE CPI

Webscraping is vrij technisch (zie hoofdstuk 4) en vergt ook een aantal programmatorische vaardigheden. Tegelijk is de benadering van de indexberekening ook anders.

Bij klassieke gegevensinzameling wordt voornamelijk gewerkt met een doelgerichte steekproef (purposive sampling). Per consumptiesegment wordt een aantal representatieve items ('getuigen') geselecteerd, waarvoor dan een definitie wordt opgesteld. Op basis van die definitie worden prijzen geregistreerd door enquêteurs. Afhankelijk van het product worden die prijzen lokaal in winkels of centraal opgenomen. Het gaat hier dus om een bottom-up benadering: er wordt eerst centraal gedefinieerd voor welke producten prijzen noodzakelijk zijn. Webscraping is meer top-down: er wordt geen voorafgaandelijke selectie toegepast. Er wordt vertrokken van de website van een retailer, boekingssite of zoekertjessite. Vrijwel alle prijzen worden ingezameld. Vervolgens worden ze geanalyseerd en geklasseerd om de noodzakelijke berekeningen te kunnen uitvoeren. Dat gebeurt eventueel met behulp van machine learning (dit wordt verder toegelicht in hoofdstuk 10).

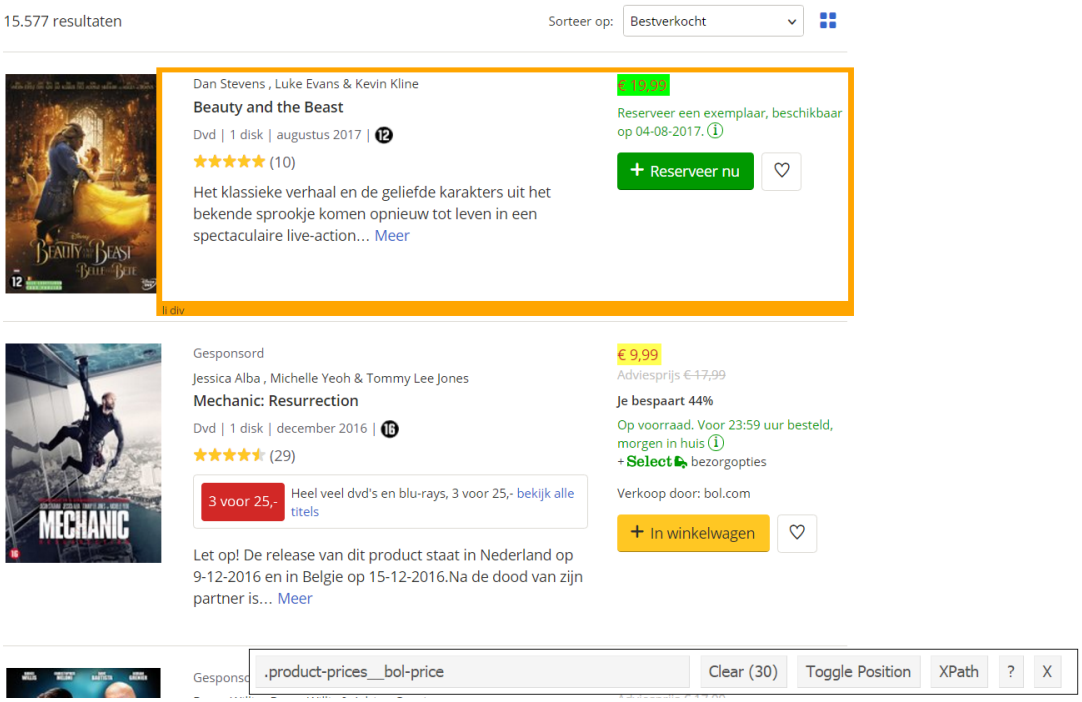
De consumptieprijsindex wordt maandelijks berekend. Wijzigingen aan een website moeten daarom snel opgemerkt worden zodat het script, dat een bepaalde site scrapet, snel kan worden aangepast. Daarom werd een dashboardapplicatie ontwikkeld die de wijzigingen kan opmerken. Verdere toelichting daarover staat in hoofdstuk 11.

# 4. WEBSCRAPING: HOE WERKT HET?

Elke website heeft een andere opmaak. Daarom worden verschillende scripts geprogrammeerd om de gegevens van de verschillende sites te kunnen afhalen. In eerste instantie moet de structuur van de website geobserveerd worden. Daarna kan er gekeken worden naar de verschillende productgroepen en eventuele subcategorieën. Als die bepaald zijn kunnen deze categorieën overlopen worden om alle overeenkomstige prijzen en productkarakteristieken te '*scrapen*'.

Statbel maakt gebruik van de opensourcesoftware en programmeertaal R om webscraping uit te voeren. Die software bevat packages die webscraping mogelijk maken: *rvest*, waarmee data van statische webpagina's gescrapet kan worden, en *RSelenium* en *phantomjs*, voor dynamische interacties met webpagina's (bijvoorbeeld het klikken op een knop, formulieren invullen, oneindig scrollen, …).

Daarnaast kan met CSS Selectors[6] of XPath de informatie die nodig is, gezocht worden in de complexe structuur van een html-pagina. Hieronder staat een voorbeeld van een CSS Selector:




**Figuur 1: Voorbeeld van CSS Selector**

De CSS Selector voor de productprijs is '.product-prices__bol-price'.

De geprogrammeerde code, om de website te bezoeken en de gegevens af te halen, wordt per website opgeslagen in 1 script. De gegevens die meestal gescrapet en dus verzameld worden, zijn: producttitel, productomschrijving, (sub-)categorie, product ID en prijs (non-promo en promoprijs). Technisch komt het er dus op neer dat via R een script wordt geprogrammeerd om virtueel een website te bezoeken en alle mogelijke interessante gegevens van de website te verzamelen en op te slaan in een zogenaamde *dataframe*.

De finale scripts/robots worden gerund via een server en identificeren zich steeds als 'Statbel'. Om de websites ook niet te overbelasten, worden bepaalde scripts 's nachts uitgevoerd en kan er geprogrammeerd worden wanneer de scripts moeten runnen (bv. dagelijks, 2x per week, wekelijks,…).

In tegenstelling tot bij manuele datacollectie, waarbij prijzen worden verzameld volgens specifieke definities, wordt bij webscraping zo veel mogelijk data verzameld. Deze data kan dan nadien verwerkt worden (eventueel met machine learning) om prijsindices te berekenen.

[6] CSS verwijst naar Cascading Style Sheets, hiermee wordt de vormgeving van elk element op een webpagina bepaald. Het element zelf wordt gedefinieerd door gebruik te maken van een HTML-tag. CSS Selectors worden gebruikt om de HTML-elementen te selecteren.

Het is uiteraard mogelijk dat niet voor alle items op een website evenveel gegevens beschikbaar zijn. Zo kan product 1 getoond worden met enkel een originele prijs, maar product 2 met een originele prijs én een promoprijs (zie figuur 2). Met de CSS-selector zien we dat voor product 1 de prijs-tag bestaat uit '.current-price'. Bij product 2 komt deze tag echter overeen met de promoprijs.

**Figuur 2: Voorbeeld van producten met verschillend aantal prijzen.**
**De geselcteerde prijzen hebben als tag 'current.price'.**

De originele prijs van product 2 heeft als tag: '.previous-price'.

**Figuur 3: De geselecteerde prijs (groen) heeft als tag '.previous-price'.**

Hier is het aan te raden om alle prijzen te verzamelen: 1 promoprijs en 2 non-promoprijzen. Bij product 1 moet er dan echter aangeduid worden dat er geen promoprijs is. Hiervoor werd een scrapefunctie ontwikkeld die een kenmerk als NA (not available) aanduidt indien het kenmerk geen waarde heeft:

```
scrape_css <- function(css,group){
                txt <- main_page %>%
                html_nodes(group) %>%
                lapply(. %>% html_nodes(css) %>% html_text() %>%
                ifelse(identical(., character(0)), NA, .)) %>%
                unlist
                return(txt)
}
```

Algemeen wordt een script opgebouwd als volgt:

- Inladen homepagina website
- Controleren op en scrapen van verschillende categorieën en subcategorieën
- Overlopen van categorieën en scrapen van kenmerken
  - Prijs
  - ID/sku
  - Productomschrijving
  - Product-url
- Samenbrengen van gegevens in dataframe (= tabel)

Voor bijvoorbeeld reiswebsites, vliegtickets of tweedehandswagens is het handiger om op voorhand een aantal criteria te definiëren. Het script zal dan enkel deze overlopen. Hierdoor worden telkens dezelfde items opgezocht en kunnen de resultaten op een correcte manier geaggregeerd en vergeleken worden. Vaak moeten op deze websites deze criteria ingevuld worden in een 'zoekfunctie'. Het script wordt dan ook zodanig geprogrammeerd dat zo'n actie wordt geïmiteerd. Het gaat dan om een virtuele boeking.

Het is ook mogelijk om bepaalde karakteristieken van een product te scrapen, zoals bijvoorbeeld bij consumentenelektronica. Hiermee kunnen hedonische prijsindices berekend worden (zie later).

De producten die op een website voorgesteld worden, staan vaak verspreid over verschillende pagina's. Het ontwikkelde script moet hiermee dus ook rekening houden en automatisch naar een volgende pagina gaan wanneer het laatste item van de voorgaande pagina gelezen is. Hiervoor kan geprogrammeerd worden dat het script op de knop '*volgende*' klikt, net zoals een gewone gebruiker dat zou doen. Er zijn echter ook websites waar meer producten geladen worden naarmate verder naar beneden gescrold wordt. Ook dat scrollen kan nagebootst worden door het script, zodat toch alle beschikbare producten gescrapet kunnen worden.

Webscraping maakt het mogelijk om op een continue manier online gegevens te verzamelen. Kwantitatief gezien wordt álle beschikbare data van de websites gescrapet (bulk scraping). Daarnaast kan er ook gericht naar data gezocht worden, op basis van vooraf bepaalde criteria, zoals dat ook manueel zou gebeuren (cfr. Robottool, hoofdstuk 12).

Op dit moment worden een 60-tal webscrapingscripts gerund voor onder andere volgende segmenten:

- Kledij
- Schoenen
- Hotels
- Vliegtickets
- Internationale treintickets
- Tweedehandswagens
- Drogisterij
- Boeken
- DVD & Blu-ray
- Videogames
- Consumentenelektronica
- Studentenkamers
- Supermarkt…

In de volgende hoofdstukken volgen een aantal experimentele resultaten van bepaalde segmenten. Aangezien het hier steeds om experimentele resultaten gaat is de tijdsaanduiding in de grafieken telkens vervangen door een nummer in plaats van de echte maand. Dat is een standaardprocedure in prijsstatistiek voor onderzoek. Zo wordt een terugkoppeling met officiële indexcijfers, met mogelijks foutieve interpretaties, vermeden. De periode verwijst telkens naar de maand. Indien niet, dan wordt dat vermeld in de tekst die de grafiek toelicht.

# 5. CASE STUDIE – CONSUMENTENELEKTRONICA

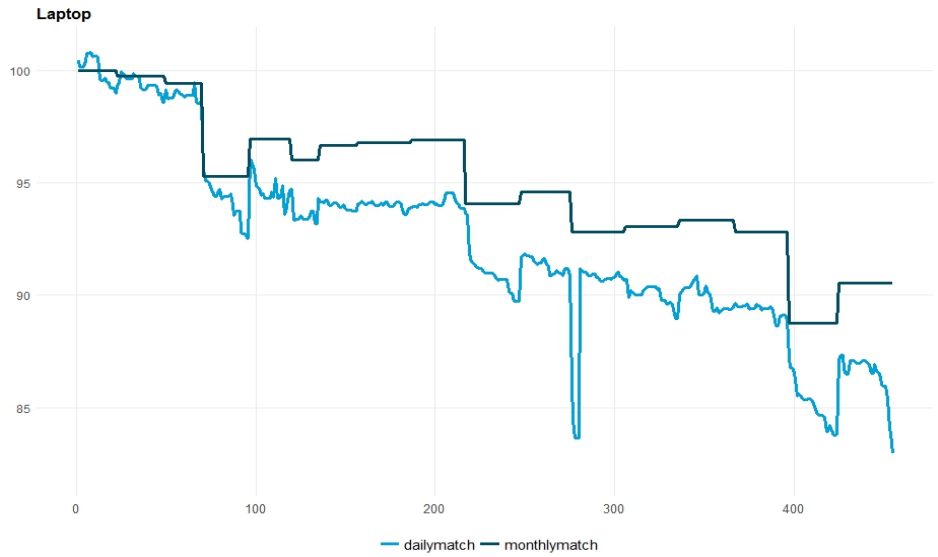
Consumentenelektronica is een domein waarvoor veel online data beschikbaar is. Gedurende minstens 17 maanden heeft Statbel data gescrapet voor verschillende elektronische toestellen, zoals laptops, tablets, koelkasten en wasmachines. Deze gegevens werden gebruikt om verschillende methoden van indexberekening te onderzoeken.

Drie factoren maken het moeilijk om prijsevoluties voor consumentenelektronica te meten:

1. Producten zijn slechts een korte tijd op de markt. Er is dus een hoge *in-flow* van nieuwe producten die gepaard gaat met een gelijktijdige *out-flow* van oude producten.
2. Deze oudere producten verlaten de markt tegen een lagere prijs (dumping of solden) dan deze waarmee ze op de markt zijn gekomen.
3. De producten die op de markt komen hebben vaak andere kenmerken: ze zijn kwalitatief beter (bijvoorbeeld energiezuiniger of in het geval van computers sneller) dan de producten die uit het gamma verdwijnen.

Deze factoren leiden tot een aantal problemen:

1. Het eerste punt zorgt ervoor dat de prijsevolutie van hetzelfde product niet gemeten kan worden voor een lange termijn. Vervangingen zijn dus vaak nodig.
2. Dat leidt ons tot het tweede punt: de prijsdaling van een elektronicaproduct tijdens haar levensduur. Als de prijsevolutie van dezelfde producten wordt gemeten op basis van een maand-op-maand Jevons kettingindex en als producten in het staal blijven tot dat zij het aanbod verlaten, zal dat leiden tot een neerwaartse kettingdrift (*downward chaindrift*). Dat blijkt uit volgende grafiek waarin een maandelijkse kettingindex en een dagelijkse kettingindex wordt weergegeven voor laptops:




**Figuur 4: Vergelijking tussen dagelijkse en maandelijkse kettingindex**

Bij de kettingindex worden nieuwe producten gekoppeld aan de tot op dat moment gemeten prijsevolutie. Door deze koppeling wordt dus het prijsverschil tussen het oude en het nieuwe product geneutraliseerd en impliciet in rekening gebracht via de prijsevolutie van de andere matchende items in de twee aanliggende maanden.

De hierboven berekende index is er dus één die berekend wordt met producten die in twee aanliggende perioden beschikbaar zijn (een *matched model* index). De formule van de Jevons-index wordt dan de volgende

$$P_J^{0,1} = \prod_{i \epsilon G_{0,1}} \left(\frac{p_i^1}{p_i^0}\right)^{1/N_{0,1}} \qquad \textit{Vgl. 1}$$

waarbij $G_{0,1}$ de steekproef is, dit is de set van alle producten die matchen in twee opeenvolgende perioden (periode 0 en

periode 1). In een kettingindex van meerdere perioden worden deze maandelijkse indexcijfers aan elkaar gekoppeld door de resulterende indexcijfers met elkaar te vermenigvuldigen. Voor een Jevons-kettingindex met 3 periodes wordt de formule dan de volgende:

$$P_J^{0,2} = \prod_{i \epsilon G_{0,1}} \left(\frac{p_i^1}{p_i^0}\right)^{1/N_{0,1}} \prod_{i \epsilon G_{1,2}} \left(\frac{p_i^2}{p_i^1}\right)^{1/N_{1,2}} \qquad \textit{Vgl. 2}$$

Uit de grafiek blijkt dat de maandelijkse kettingindex minder uitgesproken daalt. De reden hiervoor is dat in een maandelijkse aggregatie de eindprijs van het product deels afgevlakt wordt met hogere prijzen gedurende de rest van de maand. Indien telkens de lagere eindprijs van een product geïntroduceerd zou worden in het begin van de maand en het product op het einde van de maand het gamma zou verlaten, dan zijn logischerwijze de dagelijkse en maandelijkse kettingindex identiek. In de klassieke gegevensinzameling wordt getracht om het hierboven beschreven probleem van *downward chaindrift* te vermijden door te werken met *Monthly Chaining and Re-sampling* (zie later).

Hieronder volgt een theoretische uiteenzetting van verschillende indexmodellen, waarbij de modellen worden toegepast op gegevens verzameld met webscraping. Een deel van de hieronder uiteengezette modellen zal ook in andere casestudies gebruikt worden.

## 5.1. GEKS & RYGEKS

Naast het koppelen van twee aanliggende maanden bestaan er ook meer geavanceerde matched model indices zoals een GEKS-index (genoemd naar Gini, Eltetö, Köves & Szulc) waarbij niet alleen de laatste en de voorlaatste periodes telkens aan elkaar gekoppeld worden, maar ook alle tussenliggende periodes waarbij producten matchen. GEKS-indices werden ontwikkeld voor spatiale prijsvergelijkingen (gebruikt bij o.a. koopkrachtpariteiten). De bedoeling van de traditionele GEKS-procedure is dat wanneer de koopkracht van land A direct vergeleken wordt met land C, dat hetzelfde resultaat moet opleveren als wanneer die vergelijking indirect wordt gedaan door land A eerst te vergelijken met land B en land B vervolgens met land C. De GEKS-procedure is dan ook transitief in de zin dat ze onafhankelijk is van het land dat als basis wordt gebruikt. Deze GEKS-methode wordt dan voor consumptieprijsindexen logischerwijze niet voor ruimtelijke prijsvergelijkingen gebruikt, maar voor prijsvergelijkingen doorheen de tijd (temporeel).

De methode maakt gebruik van alle producten die met elkaar matchen waarbij vervolgens een ongewogen meetkundig gemiddelde (Jevons-index)[7] genomen wordt van alle mogelijke bilaterale prijsindices waarbij elke periode dient als een basisperiode. Het is ongewogen omdat er geen wegingsinformatie beschikbaar is. Een GEKS-Jevons-(of GEKS-J)index voor de periode 0 tot $t$ kan dan uitgedrukt worden als volgt:

$$GEKS_J^{0,t} = \prod_{l=0}^{T} \left({P^{0l}}/{P^{tl}}\right)^{(1/T+1)} = \prod_{l=0}^{T} (P^{0l} P^{lt})^{(1/T+1)} \qquad \textit{Vgl.3}$$

waarbij $P^{0l}$ en $P^{lt}$ bilaterale Jevons-indices zijn tussen respectievelijk de periode 0 en de periode $l$ (met 0 als basisperiode) en periode $l$ en periode $t$ (met $l$ als basisperiode) en $P^{lt} = 1/P^{tl}$. Waarbij 0, $l$ en $t$ tijdsperiodes zijn van de tijdreeks $T$ (=0, 1, …, $T$). In de bilaterale prijsindices worden enkel matched items gebruikt. In bovenstaande formule is de prijsindex van periode $t$ ook afhankelijk van de prijzen van toekomstige perioden die nog niet beschikbaar zijn ( $t$+1, $t$+2 , …) . In praktijk wordt de berekening enkel voor de meest recente periode $T$ uitgevoerd, waardoor de formule herschreven kan worden als volgt:

$$GEKS_J^{0,T} = \prod_{t=0}^{T} (P^{0t} P^{tT})^{(1/T+1)} \qquad \textit{Vgl. 4}$$

Als bij deze GEKS-J-index de periode uitgebreid wordt ($T$+1), denk bijvoorbeeld aan de volgende maand bij de CPI, dan krijgt de vergelijking volgende vorm:

[7] Voor meer informatie over deze keuze zie: Roels, D., Van Loon, K. (2017) *Het gebruik van scannerdata van supermarkten in de consumptieprijsindex*

$$GEKS_J^{0,T+1} = \prod_{t=0}^{T+1} (P^{0t} P^{tT+1})^{(1/T+2)} \qquad \textit{Vgl. 5}$$

In de GEKS-J-index worden alle bilaterale indices berekend op basis van alle matches in de dataset. Dat zorgt ervoor dat de indexcijfers voor de perioden 1, 2, …, $T$ - die berekend worden op tijdstip $T$+1 op basis van bovenstaande vergelijking - verschillen van deze die eerder berekend werden. Dit zou dan ook betekenen dat voorgaande indexcijfers telkens 'herzien' moeten worden, wat uiteraard problematisch is voor reeds gepubliceerde indexcijfers. Om dit probleem van herzieningen op te lossen kan er gewerkt worden met een *rolling year* (voortschrijdend jaar), waarbij een voortschrijdende periode van 13 maanden gebruikt wordt om indexen te berekenen. De GEKS-J wordt dan RYGEKS-J genoemd. Hierdoor moeten de resultaten niet herzien worden. Er wordt gebruik gemaakt van 13 maanden omdat dit in inflatieberekeningen overeenkomt met een jaar (bijvoorbeeld december tot december) en dit dan ook de kortst mogelijke periode is die gebruikt kan worden voor producten onderhevig aan seizoensinvloeden. In de RYGEKS-J wordt de meeste recente maand-op-maand GEKS-J-index gekoppeld aan de bestaande tijdreeks met een GEKS-J-index van de eerste 13 maanden als begin. Dit betekent dat het startpunt van de RYGEKS-J gelijk is aan:

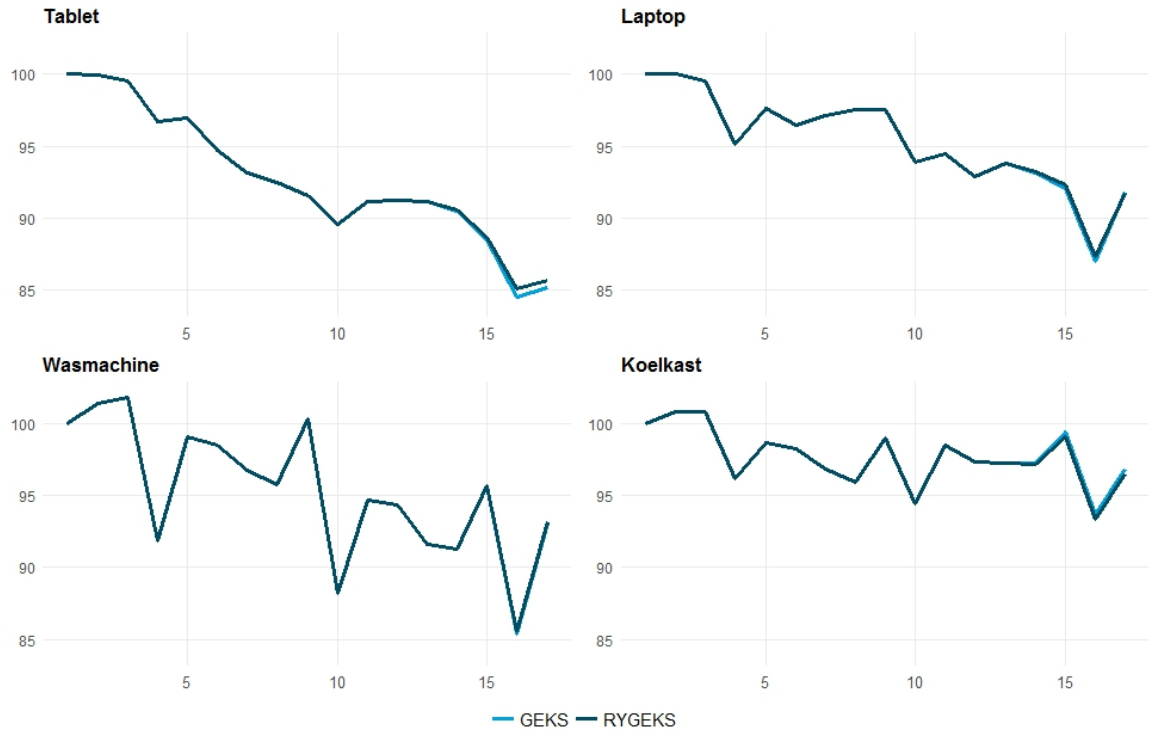
$$GEKS_J^{0,12} = \prod_{t=0}^{12} (P^{0t} P^{t12})^{(1/13)} \qquad \textit{Vgl. 6}$$

De bekomen GEKS-J-index kunnen we nu vermenigvuldigen met de meest recente maand-op-maand GEKS-J-index om de index te bekomen voor de huidige maand. De maand nadien kan de zojuist verkregen index opnieuw vermenigvuldigd worden met de nieuwe maand-op-maand index. In formulevorm levert dit het volgende op [8]:

$$\begin{aligned}
RYGEKS_J^{0,13} &= GEKS_{0,12}^{0,12} * GEKS_{1,13}^{12,13} \\
&= GEKS_J^{0,12} \prod_{t=1}^{13} (P^{12t} P^{t13})^{(1/13)} \\
RYGEKS_J^{0,14} &= RYGEKS_J^{0,13} \prod_{t=2}^{14} (P^{13t} P^{t14})^{(1/13)} \qquad \textit{Vgl. 7} \\
RYGEKS_J^{0,15} &= \cdots
\end{aligned}$$

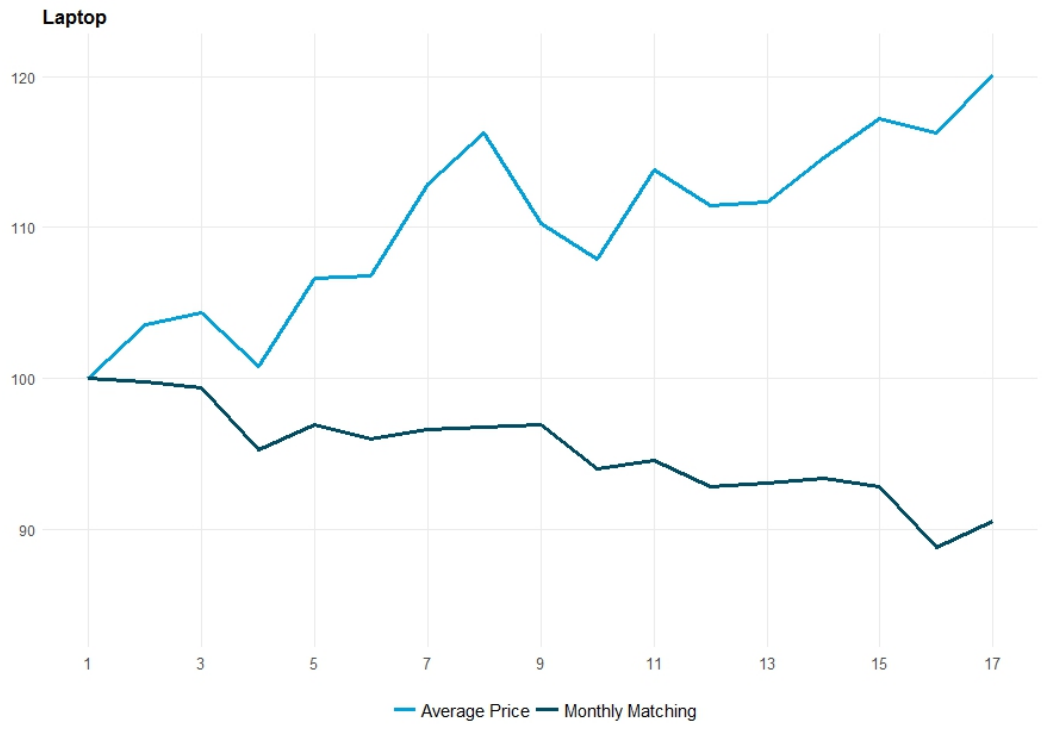
Een vergelijking tussen GEKS en RYGEKS wordt in de volgende figuur voorgesteld. Daarbij werden beide modellen toegepast op de gescrapete elektronicagegevens. Vanaf maand 14 is er een heel klein verschil waarneembaar. De eerste 13 maanden is er uiteraard geen verschil, zoals hierboven uitgelegd.

[8] De notatie $I_{c,d}^{a,b}$ staat voor de index(-methode) $I$ voor periode $a$ tot $b$ met een window over periode $c$ tot $d$. Het window geeft aan over welke periode gegevens worden gebruikt, zie ook 5.2.2. Wanneer een subscript $J$ wordt gebruikt, duidt dit op het gebruik van een Jevons-index op elementair niveau. In het vervolg van de tekst wordt dit subscript achterwege gelaten.




**Figuur 5: Vergelijking tussen GEKS en RYGEKS**

Ook het werken met een multilateraal matched model index zoals de GEKS-J lost het probleem van drift niet op, er is geen bounce back na 13 maanden. Omwille van deze drift dient het prijsverschil tussen producten die het gamma verlaten en deze die nieuw zijn in het gamma op de één of andere manier verwerkt te worden in de berekening van indexcijfers. Het rechtstreeks verrekenen van dit prijsverschil zou de inflatie dan weer opwaarts vertekenen in het geval nieuwe producten duurder zijn dan oudere producten (zie volgende grafiek). Ook zou er met het verschil in kwaliteit geen rekening gehouden worden. Onvergelijkbare producten zouden met elkaar vergeleken worden wat logischerwijze niet volgens de regelgeving van de geharmoniseerde consumptieprijsindex (HICP) is omdat het tegen één van de basisprincipes van inflatiemeting is.




**Figuur 6: Vergelijking tussen average price en monthly matching**

Eurostat stelt voor (ook in de toekomstige HICP manual) een variant van een maandelijkse kettingindex - Monthly Chaining and Re-sampling (MCR) - te gebruiken indien toch geopteerd wordt voor het werken met een maandelijkse kettingindex, aangezien een maandelijkse kettingindex nog steeds de meest toegepaste methode is met klassieke prijsopnames (dit wordt ook in België toegepast voor klassieke prijsopnames bij consumentenelektronica). Onder deze variant dienen oudere producten vervangen te worden wanneer ze nog een prijs hebben die nog representatief is en het nieuwe product dient ook opgenomen te worden in het staal tegen een representatieve prijs. Het nieuwe product wordt ook gekoppeld aan de index die tot op dat moment bestaat, d.w.z. dat het prijsverschil tussen het oude en nieuwe product niet rechtstreeks wordt verrekend, maar impliciet wordt opgevangen door de prijsevolutie van de producten die wel in de twee opeenvolgende maanden in de steekproef zijn opgenomen.

Het is niet geheel verwonderlijk dat dit moeilijk in de praktijk om te zetten is. Het is namelijk moeilijk om te bepalen wanneer de prijs van een product nu juist representatief is. Dat dit verschillende inflatiegraden oplevert voor consumentenelektronica op basis van eenzelfde methodologie is ook niet verwonderlijk. Niet alleen het moment van vervanging, maar ook het prijszettingsbeleid van winkels heeft een invloed op de gemeten prijsevolutie. Indien producten geïntroduceerd worden tegen een lagere prijs (lanceringspromo) en die prijs vervolgens licht stijgt waarna die weer daalt, zal dat een andere prijsevolutie opleveren dan wanneer een productprijs enkel een daling kent na zijn lanceringsprijs. Ook zal een prijsbeleid van graduele daling tijdens de levensduur andere inflatiecijfers opleveren dan een beleid waarbij er weinig prijsdalingen zijn tijdens de levensduur en één grote op het einde van de levensduur van het product (waar het dus vervangen dient te worden en deze daling niet in rekening dient gebracht te worden omwille van de hierboven vermelde chain drift).

## 5.2. Hedonische regressie

Een potentieel voordeel van webscraping is dat ook de karakteristieken van producten kunnen worden geregistreerd. Zo kunnen meer expliciete methoden gebruikt worden die kwaliteitswijzigingen in rekening brengen. Dat zijn de zogenaamde hedonische methoden. Die zijn moeilijk toe te passen bij klassieke prijsinzameling (online of offline). Alle karakteristieken noteren is een intensieve taak. Bovendien is het niet voldoende om enkel de kenmerken van de producten uit de steekproef te registreren bij een klassieke prijsinzameling. Dat levert te weinig observaties op om correcte regressietechnieken te kunnen toepassen.

Regressietechnieken voor consumptieprijsindices worden hedonische regressies genoemd. De prijs wordt telkens uitgedrukt als een functie van karakteristieken.

Bij een vervanging van een oud door een nieuw product is het bij hedonische methoden niet de bedoeling om op basis van het regressiemodel een raming te maken van het verschil in kwaliteit (waarna de prijs aangepast kan worden om dit verschil in rekening te brengen). Maar er wordt echter wel een raming gemaakt van wat de relatieve prijs (of de prijs) zou zijn van een nieuw product in de basisperiode of van een verdwenen product in de huidige periode.

Bovendien worden kwaliteitsaanpassingen in de gehele populatie in rekening gebracht, en niet enkel deze in de steekproef van items waar vervangingen plaatvinden. Zo zouden 1-op-1-vervangingen niet mogelijk zijn als bijvoorbeeld het aantal producten in een segment zou dalen doorheen de tijd, en er voor elk verdwenen product geen nieuwe vervanger zou zijn.

De meest toegepaste functionele vorm van hedonische regressie is een semilogaritmisch-lineair model waarbij de prijs een logaritmische transformatie ondergaat [9]. Deze logaritmische transformatie zorgt er onder andere voor dat de residuen van het model minder gekenmerkt worden door heteroscedasticiteit. De functionele vorm kan als volgt worden uitgedrukt voor een tijdreeks met twee aanliggende periodes (t=0,1):

$$\ln(p_i^t) = \alpha^t + \sum_{k=1}^{K} \beta_k^t x_{ik} + \varepsilon_i^t \qquad \textit{Vgl. 8}$$

[9] Voor meer informatie over hedonische regressie: Triplet, J. (2006), *Handbook on Hedonic Indexes and Quality Adjustments in Price Indexes*. Parijs: OECD. de Haan, J. (2010) '*Hedonic Price Indexes: A Comparison of Imputation, Time Dummy and 'Re-Pricing' Methods*', Journal of Economics and Statistics Vol. 230, No. 6, Themenheft: Index Number Theory and Price Statistics pp. 772-791.

Waar $p_i^t$ de prijs van product $i$ in periode $t$ is, waarvan vervolgens het natuurlijke logaritme genomen wordt. $x_{ik}$ staat voor de karakteristiek $k$ (k=1, …, $K$) van product $i$, met overeenkomstige parameter $\beta_k^t$. $\alpha^t$ is het intercept, de storingstermen $\varepsilon_i^t$ worden onafhankelijk verondersteld met verwachtingswaarde 0 en constante variantie.

### 5.2.1. Time Dummy Hedonics

De meest toegepaste variant van de hierboven beschreven functionele vorm is een time dummy hedonische methode waarbij de vergelijking volgende vorm aanneemt voor dezelfde context met twee aanliggende perioden ($t$=0,1):

$$\ln(p_i^t) = \alpha + \delta D_i + \sum_{k=1}^{K} \beta_k x_{ik} + \varepsilon_i^t \qquad \textit{Vgl. 9}$$

Met $D_i$ als 'time dummy' variabele waarbij deze de waarde 1 aanneemt als het product $i$ beschikbaar is in periode 1 en 0 als het niet beschikbaar is. De waarden voor de constante term $\alpha$ en de karakteristieke parameters $\beta_k$ worden verondersteld stabiel te zijn voor beide periodes ($\beta_k^0 = \beta_k^1 = \beta_k$). De storingstermen $\varepsilon_i^t$ zijn ook hier verondersteld onafhankelijk verdeeld te zijn met een verwachte waarde van nul en met gelijke variantie. De time dummy parameter $\delta$ meet het effect van tijd op het natuurlijke logaritme van de prijs, uiteraard rekening houdend met kwaliteitsbepalende karakteristieken. De time dummy index $TD_{0,1}^{0,1}$ kan dan bekomen worden door het antilogaritme (exponentiële) te nemen van $\delta$:

$$TD_{0,1}^{0,1} = \exp(\hat{\delta}) \qquad \textit{Vgl. 10}$$

Deze index kan dus direct berekend worden uit de vergelijking. Daarom wordt de time dummy hedonische methode ook vaak een directe hedonische methode genoemd. De geschatte prijzen van periode 0 ($\hat{p}_i^0$) en periode 1 ($\hat{p}_i^1$) kunnen bekomen worden uit vergelijking 9.

$$\hat{p}_i^0 = exp\left(\hat{\alpha} + \sum_{k=1}^{K} \hat{\beta}_k x_{ik}\right) \qquad \textit{Vgl. 11}$$

$$\hat{p}_i^1 = exp\left(\hat{\alpha} + \hat{\delta} + \sum_{k=1}^{K} \hat{\beta}_k x_{ik}\right) \qquad \textit{Vgl. 12}$$

Hieruit blijkt dan ook dat de time dummy index $TD_{0,1}^{0,1}$ gelijk is aan de ratio van de geschatte prijzen van alle items voor periode 1 ($\hat{p}_i^1$) ten opzichte van deze van alle items voor periode 0 ($\hat{p}_i^0$), aangezien vergelijking 10 herschreven kan worden op basis van vergelijkingen 11 en 12 is voor elk product $i$:

$$TD_{0,1}^{0,1} = \exp(\hat{\delta}) = \frac{exp\left(\hat{\alpha} + \hat{\delta} + \sum_{k=1}^{K} \hat{\beta}_k x_{ik}\right)}{exp\left(\hat{\alpha} + \sum_{k=1}^{K} \hat{\beta}_k x_{ik}\right)} = \frac{\hat{p}_i^1}{\hat{p}_i^0} \qquad \textit{Vgl. 13}$$

Aangezien de residuen sommeren tot nul in beide perioden, gelet op de opname van de constante term (en ook de time dummy), is logischerwijze het verschil tussen de som van effectieve en geraamde prijzen nul voor een set $G_0$ van beschikbare producten $i$ in periode 0 en voor een set $G_1$ van beschikbare producten $i$ in periode 1.

$$\sum_{i \in G_0} (ln\, p_i^0 - ln\, \hat{p}_i^0) = \sum_{i \in G_1} (ln\, p_i^1 - ln\, \hat{p}_i^1) = 0 \qquad \textit{Vgl. 14}$$

Wat eenvoudig herschreven kan worden als:

$$\sum_{i \in G_0} ln\left(\frac{p_i^0}{\hat{p}_i^0}\right) = \sum_{i \in G_0} ln\left(\frac{p_i^1}{\hat{p}_i^1}\right) = 0 \qquad \textit{Vgl. 15}$$

Indien nu het antilogaritme genomen wordt, resulteert dit in:

$$\exp(0) = 1 = \prod_{i \in G_0} \left(\frac{p_i^0}{\hat{p}_i^0}\right) = \prod_{i \in G_1} \left(\frac{p_i^1}{\hat{p}_i^1}\right) \qquad \textit{Vgl. 16}$$

In het geval $G_0$ = $G_1$ = $G$ is vergelijking 16:

$$\prod_{i\in G}\left(\frac{p_i^0}{\hat{p}_i^0}\right) = \prod_{i\in G}\left(\frac{p_i^1}{\hat{p}_i^1}\right) \qquad \textit{Vgl. 17}$$

Wat herschreven kan worden als:

$$\prod_{i\in G}\left(\frac{\hat{p}_i^1}{\hat{p}_i^0}\right) = \prod_{i\in G}\left(\frac{p_i^1}{p_i^0}\right) \qquad \textit{Vgl. 18}$$

Vervolgens geldt ook voor $G$ met N producten:

$$\prod_{i\in G}\left(\frac{\hat{p}_i^1}{\hat{p}_i^0}\right)^{1/N} = \prod_{i\in G}\left(\frac{p_i^1}{p_i^0}\right)^{1/N} = P_J^{0,1} \qquad \textit{Vgl. 19}$$

wat overeen komt met de Jevons index die hierboven reeds werd beschreven in vergelijking 1 ($\prod_{i\in G}\left(\frac{p_i^1}{p_i^0}\right)^{1/N} = P_J^{0,1}$).

Aangezien uit vergelijking 13 volgde dat voor alle producten $i$ de ratio $\frac{\hat{p}_i^1}{\hat{p}_i^0} = TD_{0,1}^{0,1}$ is, dan is ook logischerwijze:

$$TD_{0,1}^{0,1} = \prod_{i\in G}\left(\frac{\hat{p}_i^1}{\hat{p}_i^0}\right)^{1/N} = \prod_{i\in G}\left(\frac{p_i^1}{p_i^0}\right)^{1/N} = P_J^{0,1} \qquad \textit{Vgl. 20}$$

Waardoor de time dummy hedonische methode in de semi-logaritmische variant overeenkomt met een elementaire index formule gebaseerd op het meetkundige gemiddelde die ook in de klassieke methodologie gebruikt wordt.

In het geval $G_0 \neq G_1$, wat normalerwijze het meeste voorkomt, wordt de index berekend met items die matchen tussen de twee perioden (in overeenstemming met de matched Jevons-index omschrijven we deze set als $G_{0,1}$), verdwenen items uit de periode nul ($G_0^D$) en met nieuwe items uit periode 1 ($G_1^N$). De volledige set van items uit periode 0 is dan $G_0 = G_{0,1} \cup G_0^D$ en deze uit periode 1 is $G_1$= $G_{0,1} \cup G_1^N$. Uit vergelijkingen 11 en 12 volgt, door het nemen van het meetkundig gemiddelde over respectievelijk $G_0$ met $N_0$ producten en $G_1$met $N_1$producten:

$$\prod_{i\in G_0}(\hat{p}_i^0)^{1/N_0} = exp(\hat{\alpha})exp\left(\sum_{k=1}^{K}\hat{\beta}_k\sum_{i\in G_0}x_{ik}/N_0\right) \qquad \textit{Vgl. 21}$$

$$\prod_{i\in G_1}(\hat{p}_i^1)^{1/N_1} = exp(\hat{\alpha})\,exp(\hat{\delta})exp\left(\sum_{k=1}^{K}\hat{\beta}_k\sum_{i\in G_1}x_{ik}/N_1\right) \qquad \textit{Vgl. 22}$$

Waardoor door deling van vergelijking 22 met 21 de time dummy index gelijk is aan:

$$\begin{aligned} TD_{0,1}^{0,1} &= exp(\hat{\delta}) \\ &= \frac{\prod_{i\in G_1}(\hat{p}_i^1)^{1/N_1}exp\left(\sum_{k=1}^{K}\hat{\beta}_k\sum_{i\in G_0}x_{ik}/N_0\right)}{\prod_{i\in G_0}(\hat{p}_i^0)^{1/N_0}\,exp\left(\sum_{k=1}^{K}\hat{\beta}_k\sum_{i\in G_1}x_{ik}/N_1\right)} \\ &= \frac{\prod_{i\in G_1}(\hat{p}_i^1)^{1/N_1}}{\prod_{i\in G_0}(\hat{p}_i^0)^{1/N_0}}\,exp\left(\sum_{k=1}^{K}\hat{\beta}_k(\bar{x}_k^0-\bar{x}_k^1)\right), \qquad \textit{Vgl. 23} \end{aligned}$$

waarbij $\bar{x}_k^0 = \sum_{i \in G_0} x_{ik}/N_0$ en $\bar{x}_k^1 = \sum_{i \in G_1} x_{ik}/N_1$ en dus beide gelijk zijn aan het gemiddelde van karakteristiek $k$ in periode 0 en periode 1. Aangezien door de opname van de time dummy en de constante term ook hier de residuen sommeren tot nul in elke periode, volgt uit vergelijking 16 dat $\prod_{i \in G_0}(\hat{p}_i^0)^{1/N_0} = \prod_{i \in G_0}(p_i^0)^{1/N_0}$ en $\prod_{i \in G_1}(\hat{p}_i^1)^{1/N_1} = \prod_{i \in G_1}(p_i^1)^{1/N_1}$ . Hierdoor kan vergelijking 23 ook herschreven worden als volgt

$$TD_{0,1}^{0,1} = exp(\hat{\delta}) = \frac{\prod_{i \in G_1}(p_i^1)^{1/N_1}}{\prod_{i \in G_0}(p_i^0)^{1/N_0}} \, exp\left(\sum_{k=1}^{K} \hat{\beta}_k(\bar{x}_k^0 - \bar{x}_k^1)\right) \qquad \textit{Vgl. 24}$$

De time dummy index is dus gelijk aan de ratio van meetkundig gemiddelde prijzen (een unmatched Jevons-index) die aangepast wordt via een factor die een aanpassing doet voor de kwaliteit ($exp\left(\sum_{k=1}^{K} \hat{\beta}_k(\bar{x}_k^0 - \bar{x}_k^1)\right)$) op basis van de wijzigingen in karakteristieken tussen periode 0 en periode 1.

Als de gemiddelde karakteristieken tussen beide perioden identiek zijn ( $\bar{x}_k^0 = \bar{x}_k^1$), wat kan gebeuren als de steekproef van items identiek is of een nieuw product dezelfde karakteristieken heeft als een verdwenen product, dan is deze factor gelijk aan 1 en wordt de index dus berekend als een eenvoudige ratio van meetkundig gemiddelde prijzen. Enerzijds gelijk aan een matched Jevons-index als er *geen* wijzigingen zijn in de steekproef ($G_0 = G_1$) of anderzijds een unmatched Jevons-index als er *wel* wijzigingen zijn in de steekproef ($G_0 \neq G_1$).

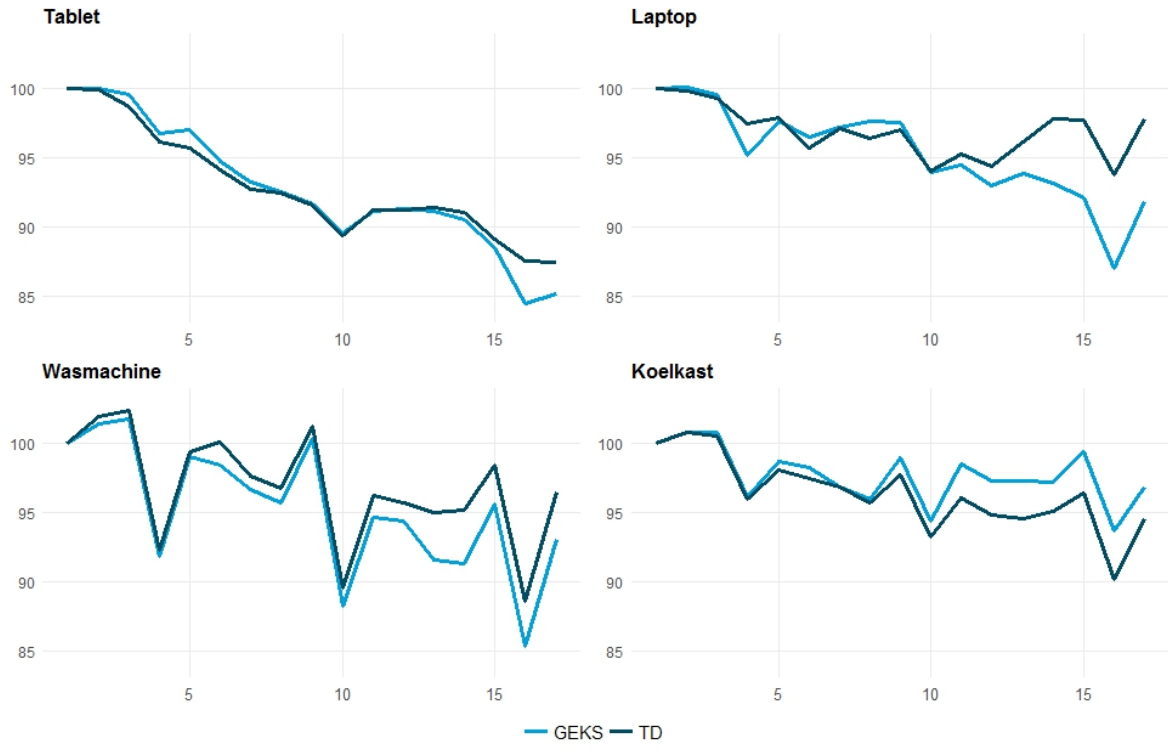
In praktijk wordt een index niet berekend over twee perioden zoals in de vergelijkingen hierboven maar over meerdere perioden $t$ = 0, …, $T$ waardoor vergelijking 9 volgende vorm aanneemt:

$$\ln(p_i^{0,T}) = \hat{\alpha} + \sum_{t=1}^{T} \hat{\delta}^t {D_i}^t + \sum_{k=1}^{K} \hat{\beta}_k x_{ik} \qquad \textit{Vgl. 25}$$

waarbij ${D_i}^t$ de 'time dummy' variabele is die de waarde 1 aanneemt als product $i$ beschikbaar is in periode $t$ en 0 als het niet beschikbaar is. Het antilogaritme (exponentiële) van $\delta^t$ geeft de time dummy index ($TD_{0,T}^{0,t}$) voor de periode $t$ (met $t = 0, 1, \ldots, T$). In een multi-periode context worden de parameters dan ook constant gehouden gedurende de hele periode. Een nadeel van deze multi-periode index is dat deze index aan herzieningen onderhevig is telkens wanneer de periode wordt uitgebreid. Voor periode $T$+1 zal het regressiemodel herschat worden, waardoor de indexcijfers voor de periode $T$ herzien zullen worden.

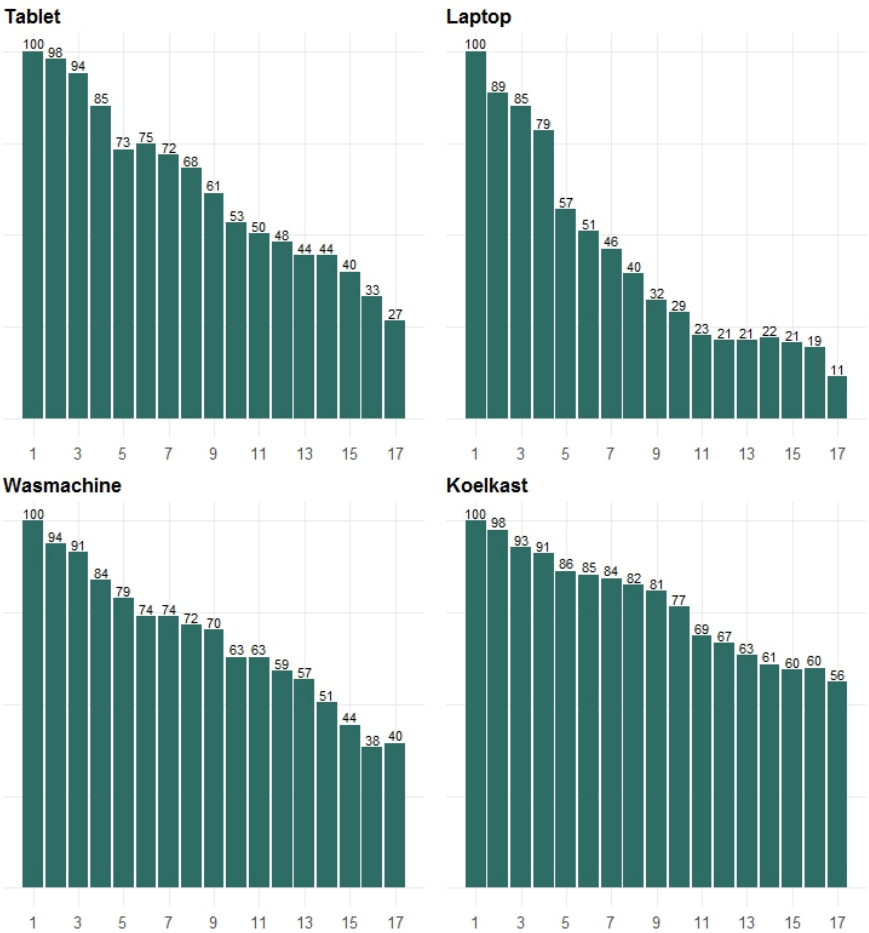
De volgende figuur geeft de resultaten voor de TD-methode vergeleken met de GEKS-methode. De selectie van de opgenomen karakteristieken in de regressievergelijking werd bepaald op basis van de analyse van het significatieniveau van de parameters, het *Akaike information criterion (AIC)* en *Bayesian information criterion (BIC)*[10]. Ook werd er gekeken naar de logica in de verhouding van de karakteristiek ten opzichte van de prijs. Zo zal bijvoorbeeld de coëfficiënt van het gewicht van een product in de regressievergelijking meestal positief zijn. Consumenten verkiezen echter ceteris paribus lichtere producten ten opzichte van zwaardere producten. Deze variabelen werden dan ook logischerwijze niet opgenomen in de regressievergelijking.

[10] AIC is een maat voor de relatieve kwaliteit van een statistisch model voor een bepaalde dataset. BIC is een criterium om een bepaald model te selecteren uit een eindige verzameling van modellen. Beide criteria zijn gelinkt met elkaar.




**Figuur 7: Vergelijking tussen Pooled Time Dummy methode en GEKS-methode**

Beide indices vertonen een gelijkaardige evolutie van prijsdaling. Bij drie van de vier producten ligt de Time Dummy index boven de GEKS-index en treedt er dus drift op wanneer de GEKS methode gebruikt zou worden. Het prijsverschil tussen nieuwe producten en producten die het gamma verlaten niet verwerken, zorgt voor een overschatting van de prijsdaling. Wanneer hiermee rekening wordt gehouden via de hedonische regressie, bekomt men een minder sterke prijsdaling. De geschatte hedonische prijsindex (TD) ligt bij het segment koelkast boven de GEKS-index. Dat is echter een vertekend beeld, om twee redenen: enerzijds is de *attrition rate* (verloop van items) voor koelkasten niet zo hoog. Dat is te zien in volgende grafiek:




Figuur 8: Attrition rate van de geselecteerde consumentenelektronica

Het aantal items dat overeenkomt is na 17 maanden nog 56%, in tegenstelling tot bij tablets (27%) of laptops (11%). Het prijsverschil tussen producten die het gamma verlaten en deze die nieuw zijn, zal dus een minder bepalende rol spelen tijdens de beschouwde periode. Anderzijds zijn er in de maand die geselecteerd werd als begin blijkbaar producten die duurder zijn dan wat men op basis van hun karakteristieken zou verwachten waardoor de hedonische prijs de hoogte ingaat. Dit wordt duidelijk als het aantal periodes wordt uitgebreid. Dat is te zien in figuur 12 (cfr. 5.2.4 Time Dummy met Window Splice). Daar valt een stijging op vanaf periode 8. Die werd in de vorige resultaten als basis genomen, waardoor het verschil geneutraliseerd werd. De verdere evolutie vertrekt dan van deze basis wat resulteert in figuur 7.

### 5.2.2. Time Dummy met Rolling Window

Analoog aan de hierboven vermelde RYGEKS kan er ook voor een time dummy hedonische index gebruikt gemaakt worden van een rolling window waarbij telkens de lengte van de gebruikte periode hetzelfde blijft, namelijk 13 maanden ($T = 0, \dots, 12$). Deze initiële rolling year time dummy (RYTD) index is dan ook gelijk aan:

$$RYTD_{0,12}^{0,12} = \exp(\hat{\delta}^{12}) \qquad \textit{Vgl. 26}$$

Met als initieel model $\hat{\alpha} + \sum_{t=1}^{12} \hat{\delta}^t {D_i}^t + \sum_{k=1}^{K} \hat{\beta}_k x_{ik}$. Waarna het model voor de volgende maand ($T$+1), geraamd wordt op basis van de perioden 1 tot 13 (i.p.v. 0 tot 12):

$$\hat{\alpha} + \sum_{t=2}^{13} \hat{\delta}^t {D_i}^t + \sum_{k=1}^{K} \hat{\beta}_k x_{ik} \qquad \textit{Vgl. 27}$$

De index voor de periode 0 tot 13 kan dan berekend worden door de meest recente periode-op-periode prijsevolutie, namelijk deze tussen maand 13 ($RYTD_{1,13}^{1,13}$) en maand 12 ($RYTD_{1,13}^{1,12}$), te koppelen aan de eerder bekomen tijdreeks tot maand 12:

$$RYTD^{0,13} = RYTD_{0,12}^{0,12} * \left( RYTD_{1,13}^{1,13} / RYTD_{1,13}^{1,12} \right) \qquad \textit{Vgl. 28}$$

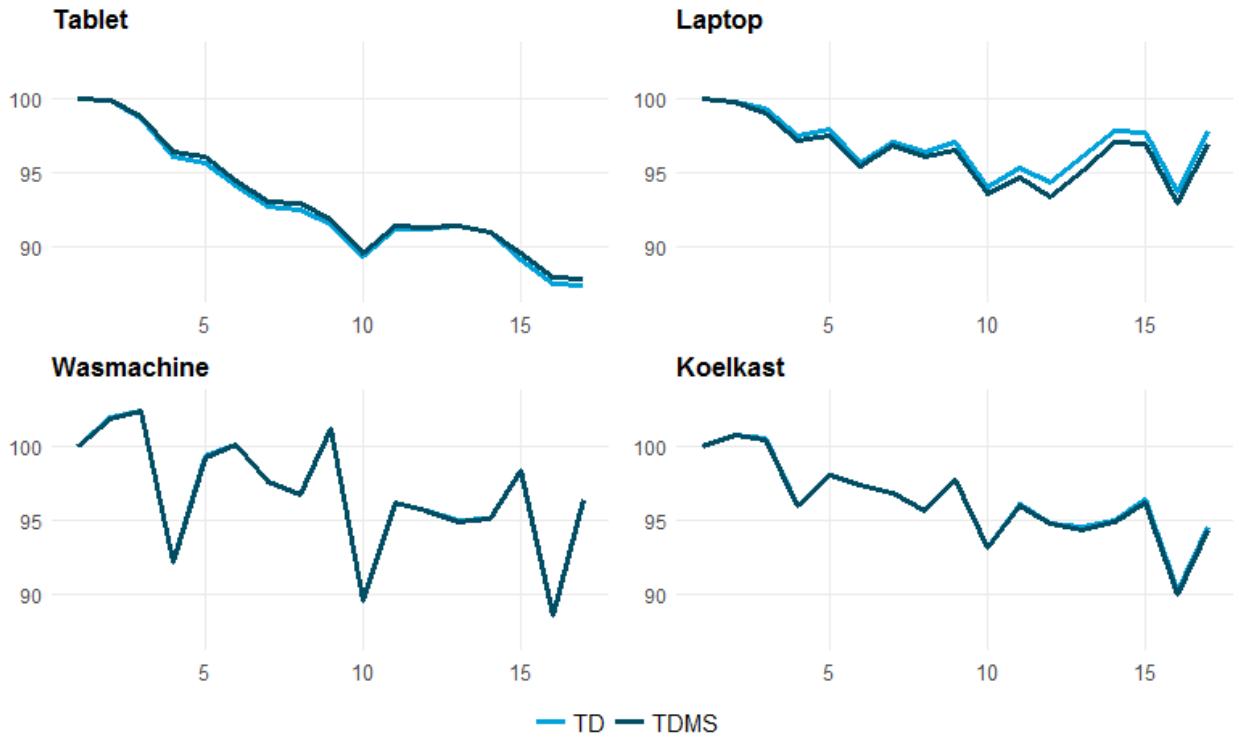
$$= RYTD_{0,12}^{0,12} * RYTD_{1,13}^{12,13}$$

Het principe voor de volgende maanden is logischerwijze identiek:

$$RYTD^{0,14} = RYTD^{0,13} * \left( RYTD_{2,14}^{2,14} / RYTD_{2,14}^{2,13} \right) \qquad \textit{Vgl. 29}$$

$$RYTD^{0,15} = \cdots$$

Omdat bij deze methode de laatste periode-op-periode (movement) gekoppeld wordt aan de bestaande tijdreeks wordt de rolling year time dummy ook wel '*Time Dummy met Movement Splice*' genoemd en wordt deze genoteerd als $TDMS^{0,T}$. Hieronder staan de resultaten voor de 4 geselecteerde segmenten voor een TDMS ten opzichte van de variant zonder rolling window (TD). Voor laptops en tablets is er een klein verschil merkbaar tussen de 'pooled' variant en de movement splice TD-index. Dat komt voornamelijk door de hogere attrition rate voor deze segmenten waardoor er meer verschillende producten in de vensterperiodes van de TDMS gebruikt worden in de raming van de regressiecoëfficiënten. Het betreft hier uiteraard een vrij korte tijdsperiode en het verschil zal groter worden naarmate de tijdsperiode wordt uitgebreid.




**Figuur 9: Vergelijking tussen Pooled Time Dummy methode en Time Dummy met Movement Splice**

Wanneer geen informatie over de karakteristieken beschikbaar is, kan een hedonische methode niet worden toegepast. In de standaard vergelijking voor een time dummy index $\left(\hat{\alpha} + \sum_{t=1}^{T} \hat{\delta}^t D_i{}^t + \sum_{k=1}^{K} \hat{\beta}_k x_{ik}\right)$ ontbreken dan de hedonische effecten $\left(\sum_{k=1}^{K} \hat{\beta}_k x_{ik}\right)$.

### 5.2.3. Fixed Effects met Window Splice

Krsinich (2016)[11] stelt dat toch kwaliteitscorrecties kunnen gebeuren op basis van zogenaamde product specifieke '*fixed effects*' $\gamma_i$ die de ontbrekende hedonische effecten vervangen. De functionele vorm voor een fixed effects model ziet er dan voor elk product $i$ als volgt uit:

$$\ln(p_i^{0,T}) = \hat{\alpha} + \sum_{t=1}^{T} \hat{\delta}^t {D_i}^t + \sum_{i=1}^{N-1} \hat{\gamma}_i D_i \qquad \textit{Vgl. 30}$$

waarbij ${D_i}^t$ een 'time-dummy'-variabele is die de waarde 1 aanneemt als product $i$ beschikbaar is in periode $t$ en 0 als het niet beschikbaar is, $D_i$ is een 'product-dummy' die de waarde 1 aanneemt als de observatie gerelateerd is aan product $i$ . Het antilogaritme (exponentiële) van $\hat{\delta}^t$ geeft de fixed effects index ($FE^{0,t}$) voor de periode $t$. Op basis van een analoge redenering die opgebouwd werd in vergelijking 10 tot 24 kan een fixed-effects model ook geschreven worden als volgt:

$$FE^{0,T} = exp(\hat{\delta}^T) = \frac{\prod_{i \in G_T} (p_i^T)^{1/N_T}}{\prod_{i \in G_0} (p_i^0)^{1/N_0}} \, exp(\bar{\gamma}^0 - \bar{\gamma}^T) \qquad \textit{Vgl. 31}$$

waarbij $\bar{\gamma}^0 = \sum_{i \in G_0} \hat{\gamma}_i / \ N^0$ en $\bar{\gamma}^T = \sum_{i \in G_T} \hat{\gamma}_i / \ N^T$ de gemiddelden zijn van de geraamde fixed effects.

Een item moet minstens twee perioden beschikbaar zijn om een relevante impact te hebben op een fixed effects index (in tegenstelling tot bij een time dummy hedonische index waarbij deze meteen een effect kan hebben). Fixed effects worden namelijk berekend op basis van een fixed effects regressie. Die maakt gebruik van paneldata (meerdere observaties van dezelfde variabele over meerdere tijdsperioden) waardoor enkel producten die minstens twee perioden beschikbaar zijn een impact hebben op de regressie (en logischerwijze de daaruit volgende fixed effects indices)[12].

Vanaf de tweede periode waarin het product beschikbaar is, zal het fixed effect van dat product wel een impact hebben op de daaruit volgende fixed effects index. Hoe meer observaties (= perioden) beschikbaar zijn van een product, hoe meer het fixed effect van het product convergeert naar haar werkelijke waarde. Dat zorgt ervoor dat de hierboven beschreven methode van het werken met een rolling year niet aangewezen is. Enkel de meest recente periode-op-periode evolutie wordt gekoppeld aan de reeds bestaand indexreeks, waardoor de geüpdatete waarde van de fixed effects raming van het product in kwestie niet verwerkt wordt in vroegere indexcijfers.

Om dat probleem op te lossen, stelt Krsinich voor om te werken met een *window splicing* methode. Deze methode wordt FEWS (fixed effects with window splicing) genoemd. De hele nieuwe 13-maanden-raming wordt gekoppeld (in het geval $T$ telkens 13 maanden is) aan de index van 12 maanden geleden, die zelf bekomen werd uit de raming van de vorige 13 maanden. Op die manier worden de geüpdatete waarden van de fixed effects wel verwerkt in het indexcijfer. De FEWS-index voor de eerste 13 maanden wordt standaard berekend zoals voor de time dummy index:

$$ln(p_i^{0,12}) = \hat{\alpha} + \sum_{t=1}^{12} \hat{\delta}^t {D_i}^t + \sum_{i=1}^{N-1} \hat{\gamma}_i D_i \qquad \textit{Vgl. 32}$$

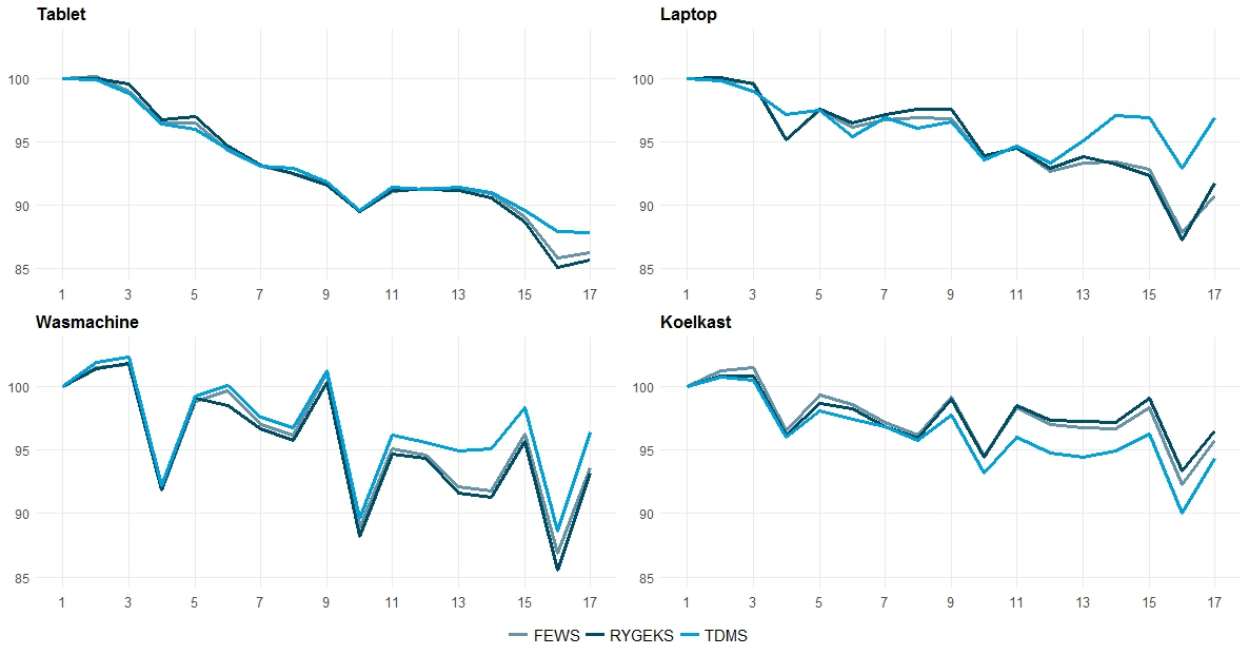
Voor de 14$^{de}$ maand wordt de index ($FEWS^{0,13}$) berekend door de prijsevolutie van de nieuwe ramingsperiode van 13 maanden te koppelen aan de prijsindex van 12 maanden geleden:

$$FEWS^{0,13} = FEWS_{0,12}^{0,12} * \frac{FEWS_{1,13}^{1,13}}{100} \qquad \textit{Vgl. 33}$$

[11] Krsinich, F. (2016). *The FEWS Index: Fixed Effects with a Window Splice*. Journal of Official Statistics, Vol.32, No. 2, 2016, pp. 375-404.

[12] Wanneer de index berekend wordt met slechts twee perioden zijn de gemiddelde geraamde effecten in beide perioden $\bar{\gamma}^0$ en $\bar{\gamma}^1$ identiek, waardoor het verschil tussen deze twee gemiddelden nul is en de fixed effects index identiek is aan een matched Jevons-index.

In volgende grafiek wordt de FEWS-index vergeleken met de RYGEKS en de TDMS. Er kan opgemerkt worden dat de resultaten van de FEWS-methode zeer weinig verschillen met deze van de RYGEKS en dat de 'fixed effects' toch niet de hedonische effecten kunnen vervangen. Hedonische methoden zonder karakteristieken lijken dus moeilijk realiseerbaar. Dat is ook wat de Haan en Hendriks (2013) constateerden[13].




**Figuur 10: Vergelijking tussen FEWS- en RYGEKS-methode en Time Dummy met Movement Splice**

### 5.2.4. Time Dummy met Window Splice

De hierboven vermelde window splice methode kan ook toegepast worden op de time dummy hedonische methode, dit wordt dan Time Dummy met Window Splice (TDWS) genoemd. Net zoals de Time Dummy met Movement Splice (TDMS) maakt de TDWS gebruik van een rolling window en kan het ook algemeen genoteerd worden als $TDWS^{0,T}$. De manier waarop de opeenvolgende ramingsperioden gekoppeld worden, verschilt echter.

De index voor de eerste 13 maanden is in beide gevallen wel identiek:

$$\ln(p_i^{0,12}) = \hat{\alpha} + \sum_{t=1}^{12} \hat{\delta}^t {D_i}^t + \sum_{k=1}^{K} \hat{\beta}_k x_{ik} \qquad \textit{Vgl. 34}$$

Ook de raming van de volgende 13 maanden is identiek.

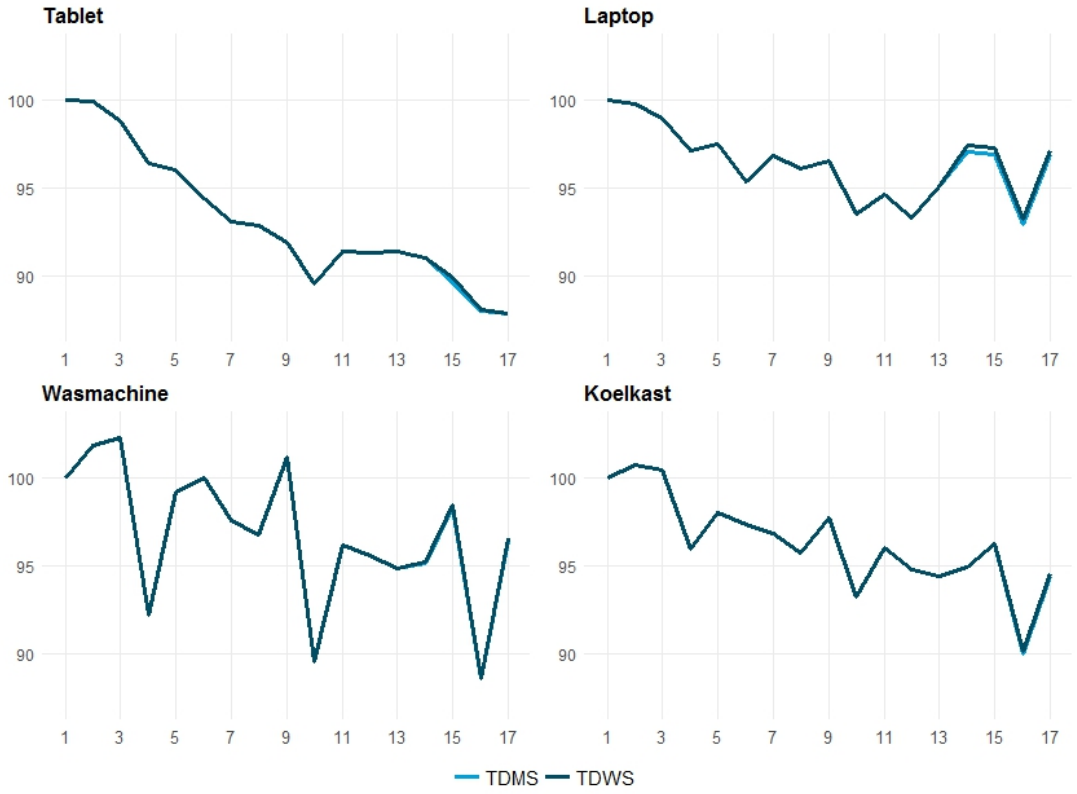
$$\ln(p_i^{1,13}) = \hat{\alpha} + \sum_{t=1}^{12} \hat{\delta}^t {D_i}^t + \sum_{k=1}^{K} \hat{\beta}_k x_{ik} \qquad \textit{Vgl. 35}$$

Het verschil tussen beide rolling window methoden is hoe de index van de 14$^{de}$ maand wordt berekend. Ter herinnering, in de movement splice wordt enkel de maand-op-maand evolutie van de laatste tweede maanden in de nieuwe ramingsperiode gekoppeld aan de oude index (zie vergelijking 28). Terwijl in de window splice methode de hele nieuwe 13 maanden raming gekoppeld wordt (in het geval $T$ telkens 13 maanden is) aan de index van 12 maanden geleden die bekomen werd uit de raming van de vorige 13 maanden:

$$TDWS^{0,13} = TDWS_{0,12}^{0,12} * \frac{TDWS_{1,13}^{1,13}}{100} \qquad \textit{Vgl. 36}$$

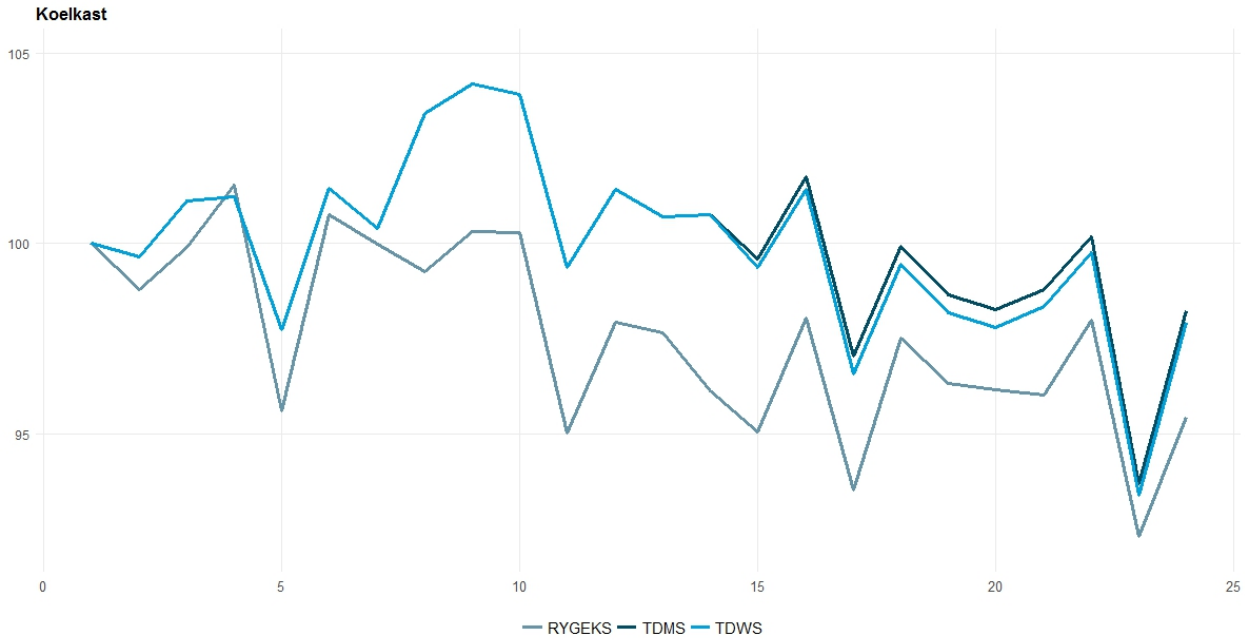
[13] de Haan, J., Hendriks, R. (2013), *Online Data, Fixed Effects and the Construction of High-Frequency Price Indexes*, Economic Measurement Group, Sydney, Australië.

Dat betekent dat reeds gepubliceerde indexcijfers in beide gevallen ongewijzigd blijven, en voor de eerste 13 maanden zelfs identiek zijn tussen beide methoden. Vanaf de 14$^{de}$ maand is het indexcijfer normalerwijze verschillend omdat de window splice methode voor de periode $T$+1 (=0,…, $T$, $T$+1) de prijsevolutie op basis van nieuwe parameters tussen maand 1 en $T$+1 in rekening brengt, terwijl in de movement splice methode enkel de prijsevolutie op basis van de nieuwe parameters tussen $T$ en $T$+1 in rekening wordt gebracht.

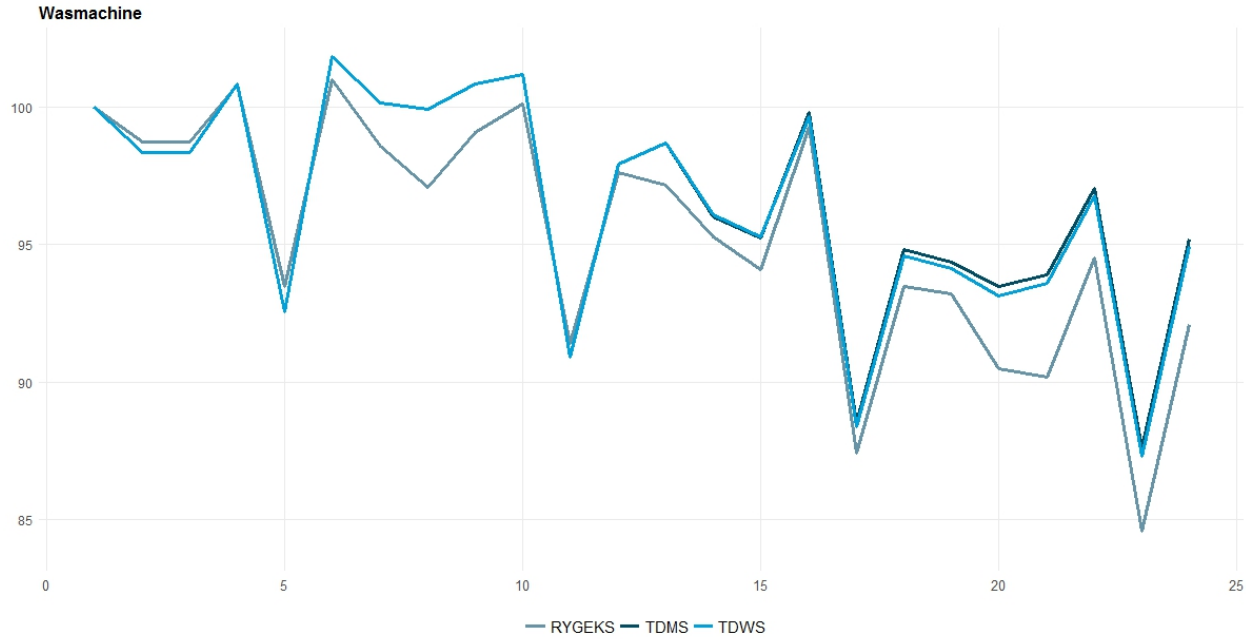



**Figuur 11: Vergelijking tussen Time Dummy met Movement Splice en Time Dummy met Window Splice**

De bovenstaande figuur toont een klein verschil tussen de time dummy methode met movement splice en de methode met window splice. Aangezien de prijsevolutie in de eerste 13 maanden gelijk is, zien we enkel een verschil vanaf maand 14. Wanneer over een langere periode (24 maanden) gegevens verzameld en verwerkt worden, is het verschil een klein beetje duidelijker. Bovendien bevinden de TD-indices zich voor de koelkast  boven de RYGEKS:




**Figuur 12: Vergelijking RYGEKS, TDMS en TDWS voor koelkasten over een periode van 24 maanden**




**Figuur 13: Vergelijking RYGEKS, TDMS en TDWS voor wasmachines over een periode van 24 maanden**

Het nadeel van de window splice methode is dat de interpretatie van de maand-op-maand evolutie tussen $T$ en $T$+1 moeilijker is, omdat deze berekend wordt op basis van twee ramingsperioden 0 tot $T$ en 1 tot $T$+1 en dit op basis van twee modellen en dus niet de maand op maand prijsevolutie weergeeft van éénzelfde model zoals in de movement splice. De window splice methode biedt echter wel het voordeel dat de impact van gewijzigde parameters over de ganse periode verrekend wordt. Een ander voordeel is dat, wanneer een nieuwe karakteristiek geïntegreerd wordt in het regressiemodel, het effect van die toevoeging op de hele vensterperiode wordt verrekend en niet enkel het effect hiervan op de laatste twee maanden zoals met de movement splice het geval is. Hierdoor gaat momenteel de voorkeur uit naar de Time Dummy met Window Splice-methode.

## 5.3. Bedenkingen bij hedonische methoden

Hedonische methoden worden momenteel bijna nergens in Europa toegepast. In de eurozone worden ze momenteel enkel toegepast in Duitsland voor tweedehandswagens, tablets en computers, in Frankrijk voor boeken, wasmachines en koelkasten en in Nederland enkel voor tweedehandswagens. Eén van de grootste problemen voor de toepassing van hedonische methoden is dat het vrij veel data vereist en dus arbeidsintensief is ten opzichte van het belang van deze producten in de indexkorf.

Niet alleen moeten karakteristieken van de producten genoteerd worden, ook de steekproef moet voldoende groot zijn. Dat is belangrijk om voldoende vrijheidsgraden te garanderen, en zo tot betrouwbare ramingen te komen voor de parameters in de regressievergelijking. Hierdoor wordt de gegevensinzameling vaak uitbesteed voor die productgroepen (bijvoorbeeld in Duitsland en Nederland), of zelfs de hele berekening van de index in het geval van de index voor tweedehandswagens in Nederland.

Via webscraping wordt het wat makkelijker om voldoende observaties te hebben en om de karakteristieken te registeren. Deze gegevens verwerken en de scripts schrijven voor de scraping van de websites is uiteraard ook arbeidsintensief. Ook deze methode kan dus slechts voor een aantal productgroepen worden toegepast.

De variabelen zijn vaak niet continu verdeeld (slechts een aantal waarden per variabele, en ze zijn zeker niet het resultaat van een Gaussiaans stochastisch proces). Denk bijvoorbeeld aan waterverbruik of elektriciteitsverbruik voor een wasmachine, waarbij het op een bepaald moment telkens een discrete verdeling heeft met kleine variantie. Een grotere variantie per variabele wordt slechts bekomen over een grotere tijdsperiode waarbij de oorspronkelijke regressievergelijking minder representatief wordt. Hier biedt webscraping wel een voordeel ten opzichte van hedonische methoden die gebruikt werden met klassieke gegevensinzameling. Daar werd vaak de vergelijking slechts sporadisch geraamd omwille van het arbeidsintensieve proces van het inzamelen van voldoende prijzen en karakteristieken. Met webscraping is het mogelijk om een regressievergelijking te hanteren die continu up-to-date is. Regelmatig gegevens webscrapen zorgt voor steeds uitgebreidere en meer accurate data. De scripts werden ondertussen ook nog aangevuld met nieuwe segmenten zoals home cinema’s, TV’s en smartphones.

# 6. CASE STUDIE – TWEEDEHANDSWAGENS

Tweedehandswagens is een segment dat momenteel nog niet vervat zit noch in de geharmoniseerde, noch in de nationale consumptieprijsindex. Het gewicht ervan bevindt zich nochtans boven de 1-promille-drempel die beschreven wordt in de HICP-regelgeving.

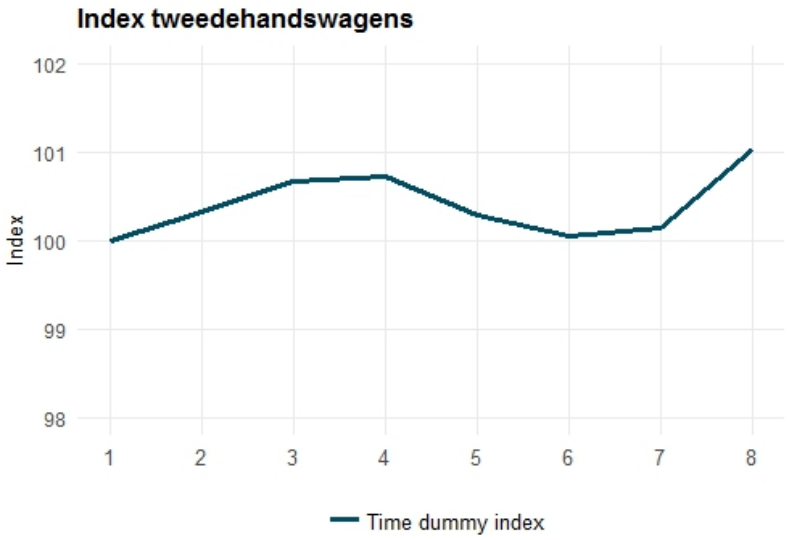
Deze regelgeving stelt dat alle consumptieve goederen en diensten waarvoor de consumptieve bestedingen van huishoudens zich boven deze drempel bevinden, verplicht opgenomen moeten worden in de berekening van de HICP. Dit consumptiesegment was tot op heden door Statbel niet opgenomen. Het probleem daarvoor ligt bij de gegevens: tot nu toe konden geen gegevens verkregen worden die een correcte berekening van een index voor tweedehandswagens mogelijk maakt.

Voor dit segment kan ook geen index berekend worden die op één of andere manier niet corrigeert voor het verschil in aangeboden tweedehandswagens. De leeftijd, het aantal kilometer, type van brandstof, … is voor een bepaald type van wagen telkens anders.

Via webscraping wordt het eenvoudiger om deze index te berekenen: voldoende prijzen (alsook karakteristieken) zijn beschikbaar op sites die zich specialiseren in het aanbieden van tweedehandswagens. Een steekproef van de meest populaire tweedehandswagens werd opgesteld op basis van de database met inschrijvingen van auto's van de Federale Overheidsdienst Mobiliteit. Nadien werden voor deze steekproef vrijwel dagelijks gegevens gescrapet. Prijzen van eenzelfde observatie die meerdere malen voorkwam, werden herleid tot één prijs.

Na het bijwerken van de gegevens (verwijderen van bizarre prijzen, leeftijd van de wagen, aantal kilometer, …) werd de index geraamd op basis van de eerder beschreven hedonische time dummy index. Ook hier werd de semi-logaritmische variant gebruikt. De gegevens van de verschillende maanden werden gebruikt in de regressievergelijking, waarbij elke maand (=periode) een dummy kreeg (het gaat hier dus om de pooled regressie variant).

De bekomen index wordt weergegeven in de volgende grafiek, waaruit vastgesteld kan worden dat de opgemeten prijsevolutie vrij stabiel is. Dat is niet geheel onlogisch. Het zou zelfs verwonderlijk zijn dat voor 'gestandaardiseerde' wagens de prijs sterk zou fluctueren van maand op maand.




**Figuur 14: Time Dummy index voor tweedehandswagens**

Het is aangewezen om na te gaan hoe de time dummy index exact corrigeert voor kwaliteitsverschillen tussen wagens. Uit het vorige hoofdstuk blijkt dat dit type van index voor een bepaalde periode opgesplitst kan worden in een ongecorrigeerde index (ratio van meetkundig gemiddelde prijzen), waarbij met kwaliteitsverschillen van de observaties tussen perioden geen rekening wordt gehouden, en een factor die een aanpassing doet voor de kwaliteitsverschillen tussen perioden op basis van de wijzigingen in karakteristieken. De gestandaardiseerde formule voor periode $t$ hiervoor is:

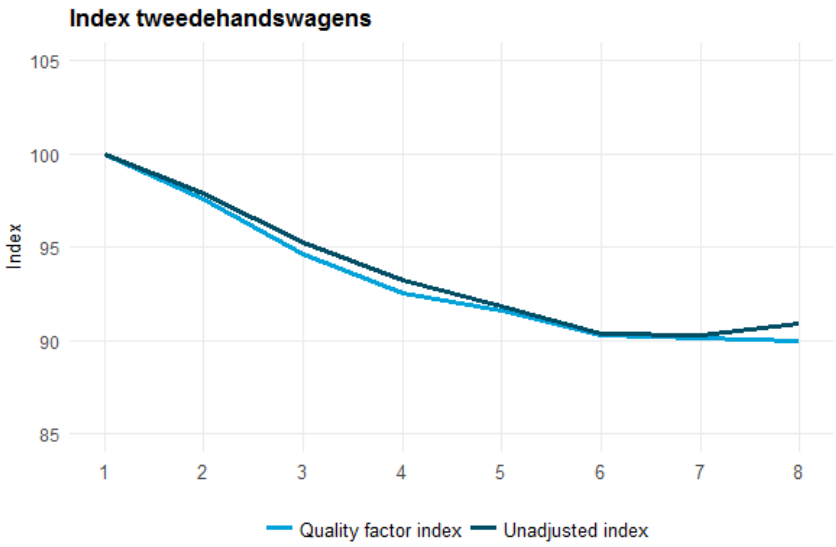
$$TD^{0,t} = exp(\hat{\delta}) = \frac{\prod_{i \in G_t}(p_i^1)^{1/N_1}}{\prod_{i \in G_0}(p_i^0)^{1/N_0}} \, exp\left(\sum_{k=1}^{K} \hat{\beta}_k(\bar{x}_k^0 - \bar{x}_k^t)\right) \qquad \textit{Vgl. 37}$$

Deze kan herschreven worden als volgt, waarbij simpelweg de inverse genomen wordt van de tweede component in de vergelijking hierboven om een ratio te bekomen in plaats van een vermenigvuldiging:

$$\begin{aligned} TD^{0,t} &= \frac{\prod_{i \in G_t}(p_i^1)^{1/N_1}}{\prod_{i \in G_0}(p_i^0)^{1/N_0}} \Bigg/ \exp\left(\sum_{k=1}^{K} \hat{\beta}_k(\bar{x}_k^0 - \bar{x}_k^t)\right)^{-1} \\ &= \frac{\text{Unadjusted index}}{\text{Quality factor index}} \end{aligned}$$

*Vgl. 38*

Hierdoor wordt $TD^{0,t}$ dus gelijk aan de ratio van 2 indexcijfers, namelijk deze van een Jevons-index (*unadjusted index*) en een index voor de kwaliteit van de wagens in het staal (*quality factor index*). Op deze manier wordt een type van 'nominale indicator' bekomen die gecorrigeerd wordt door een deflator. De opsplitsing tussen beide indexcijfers wordt weergegeven in volgende grafiek.




**Figuur 15: Unadjusted en Quality factor index voor tweedehandswagens**

Beide indexcijfers kennen een daling. Dat wil niet alleen zeggen dat de gemiddelde prijzen in de steekproef van de aangeboden wagens gedaald zijn, maar ook dat de kwaliteit van de aangeboden wagens gedaald is. Een deling van beide indexcijfers is gelijk aan de time dummy index. In periode 8 wordt bijvoorbeeld vastgesteld dat de quality factor index sterker is gedaald dan de unadjusted index van meetkundig gemiddelde prijzen waardoor de time dummy index een opwaartse beweging kent naar een niveau van 101.

Gelet op de bekomen resultaten en de regelgeving voor de HICP, zal dit consumptiesegment normaal gezien opgenomen worden vanaf 2019. Er dient dan ook een keuze gemaakt te worden tussen window of movement splicing (of een andere splicing methode die buiten de reikwijdte van deze tekst valt). Voor de nationale consumptieprijsindex is dit afhankelijk van het advies van de Indexcommissie.

# 7. CASE STUDIE – HOTELS

Voor hotelkamers worden sinds een aantal maanden gegevens gescrapet voor drie Belgische bestemmingen: een weekend aan de Belgische kust, een weekend in de Ardennen en een weekend in Brussel.

Bij de manuele prijsopnames is een steekproef van hotels opgemaakt en worden de prijzen 1 keer per maand verzameld voor een boeking 4 weken voor de aankomstdatum (1 prijsopname per hotel). De virtuele boeking wordt gemaakt voor een 2-persoonskamer, voor een verblijf van twee nachten. In principe worden het kamertype en andere opties stabiel gehouden. Uiteraard is dit technisch niet altijd haalbaar aangezien dit afhankelijk is van de beschikbaarheid van de kamers. De steekproef bestaat uit hotels voor Brussel, voor de Belgische kust en voor de Ardennen.

Bij het webscrapen worden dagelijks de gegevens verzameld voor een verblijf waarbij 4 én 8 weken voor aankomst (virtueel) geboekt wordt. Dit levert 1 prijs op per hotel per boekingsdatum voor een verblijf 4 en 8 weken na de boekingsdatum. De virtuele boekingen worden hier steeds uitgevoerd voor een aankomst op vrijdag en vertrek op zondag en omvatten ook ontbijt en gratis annulatie. Verder wordt een extra stratificatie gemaakt per streek (Kust en Ardennen), zo zitten er 5 kuststeden en 9 Ardense bestemmingen in de steekproef. De Brusselse hotels worden beperkt tot het stadscentrum. Daarna is er ook nog een stratificatie op sterquotering: 2, 3 en 4 sterren. De volledige stratificatie kan visueel voorgesteld worden als:

Streek

↓

Bestemming

↓

Aantal weken op voorhand geboekt

↓

Aantal sterren

Er dient verduidelijkt te worden dat, in overeenstemming met de HICP-regelgeving rond de berekening van indexcijfers voor diensten, de gemeten prijsevolutie verwerkt wordt in het indexcijfer van de maand waarin de dienst effectief begint. Voor een verblijf in een hotel wil dit dus zeggen dat de dag waarin volgens de boeking kan worden ingecheckt in het hotel zal bepalen in welke maand de prijs verrekend wordt in het indexcijfer. De redenering hierachter is dat de maand van verblijf meer bepalend is voor de prijsevolutie dan het moment van boeking, m.a.w. de gemeten prijsevolutie tussen maanden onderling hangt meer af van de maand van verblijf dan van hoeveel weken op voorhand geboekt is. Concreet zal de prijs van een boeking voor een verblijf dat start in april, ongeacht of deze nu 4 of 8 weken op voorhand gebeurde, verwerkt worden in het indexcijfer van april. In het indexcijfer van april zullen dus prijzen van boekingen verwerkt worden die gebeurd zijn in februari, maart of april zelf. Ook is de vertrekdatum irrelevant, een verblijf dat start op 30 april en eindigt op 2 mei zal verwerkt zijn in het indexcijfer van april.

Op het laagste niveau van het hierboven schematisch weergegeven stratificatiemodel wordt er een Jevons-index gebruikt. Vervolgens wordt de meetkundig gemiddelde prijs op het laagste niveau telkens afgezet tegen de prijs zoals geregistreerd in de basismaand (eerste maand). De resulterende indexen worden vervolgens stapsgewijs geaggregeerd naar een hoger niveau om op die manier een index te bekomen per streek. Er wordt via webscraping dus niet langer een prijs per hotel berekend, maar per stratum.

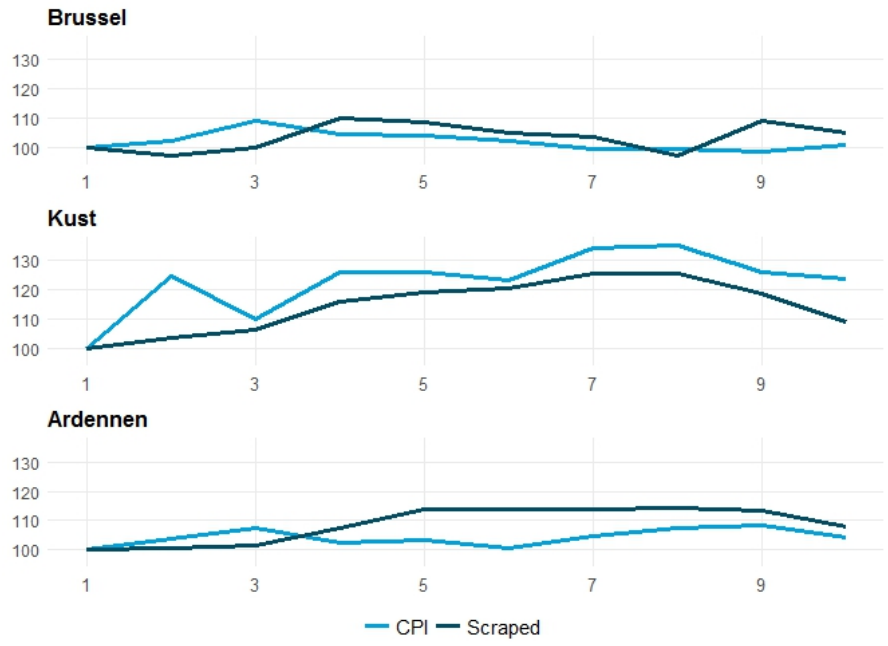
De strata, samen met het beperken van de boeking tot een bepaald kamertype met dezelfde opties, alsook de periode van verblijf, leiden er toe dat een homogene dienst bekomen wordt op het laagste niveau, waarbinnen geen kwaliteitsaanpassingen nodig zijn. Door het werken met deze methode is er ook altijd een prijs per stratum beschikbaar, net als een ‘basisprijs’ per stratum die berekend kan worden in december waartegen de prijzen in het lopende jaar kunnen worden afgezet. Hierdoor is er geen methodologisch probleem om prijzen te verwerken van hotels waarvan er in een

bepaalde periode geen kamers beschikbaar zijn of het verwerken van prijzen van nieuwe geopende hotels. Werken met een rolling year is hier dan ook niet noodzakelijk.

Via webscraping is het ook eenvoudig om het aantal hotels in de steekproef te verhogen. Door het dagelijkse scrapen is het aantal prijsobservaties dan uiteraard ook veel hoger dan bij de manuele methode. Het aantal gescrapete prijzen voor 1 maand is gemiddeld:

| Regio | Aantal prijzen |
|---|---|
| Brussel | 2.662 |
| Kust | 12.614 |
| Ardennen | 23.552 |

Dit alles leverde volgende resultaten:




**Figuur 16: Vergelijking van indexevolutie tussen data via webscraping en CPI-cijfers**

De gescrapete data levert globaal gezien eenzelfde trend in indexevolutie als de officiële CPI-cijfers. De resultaten bij Brussel zijn zelfs logischer aangezien de paasvakantie in 2017 volledig in april (maand 4) viel, zodat er een indexverhoging verwacht werd in april. De officiële cijfers tonen echter een stijging in maart (maand 3) en een daling in april (maand 4).

Ondertussen werd het aantal bestemmingen waarvoor gegevens gescrapet worden nog opgevoerd met 5 Vlaamse steden en 7 Waalse. Verder onderzoek staat gepland in 2018.

De gescrapte indices leveren een goed resultaat en lijken minder volatiel dan de officiële indices. Dat is mogelijks te verklaren door de grotere steekproef, de stratificatie en het beperken van de boeking tot kamers met ontbijt en gratis annulatie. Aangezien bij de manuele methode slechts 1 keer per maand de prijs wordt opgenomen voor hetzelfde hotel is het mogelijk dat een bepaald kamertype niet meer beschikbaar is op het moment van boeking. Hierdoor zal dan een andere kamer gekozen worden in hetzelfde hotel of eventueel dezelfde kamer met andere voorwaarden (bijvoorbeeld dezelfde kamer waar de gratis annulatie mogelijks niet inbegrepen is).

# 8. CASE STUDIE – STUDENTENKAMERS

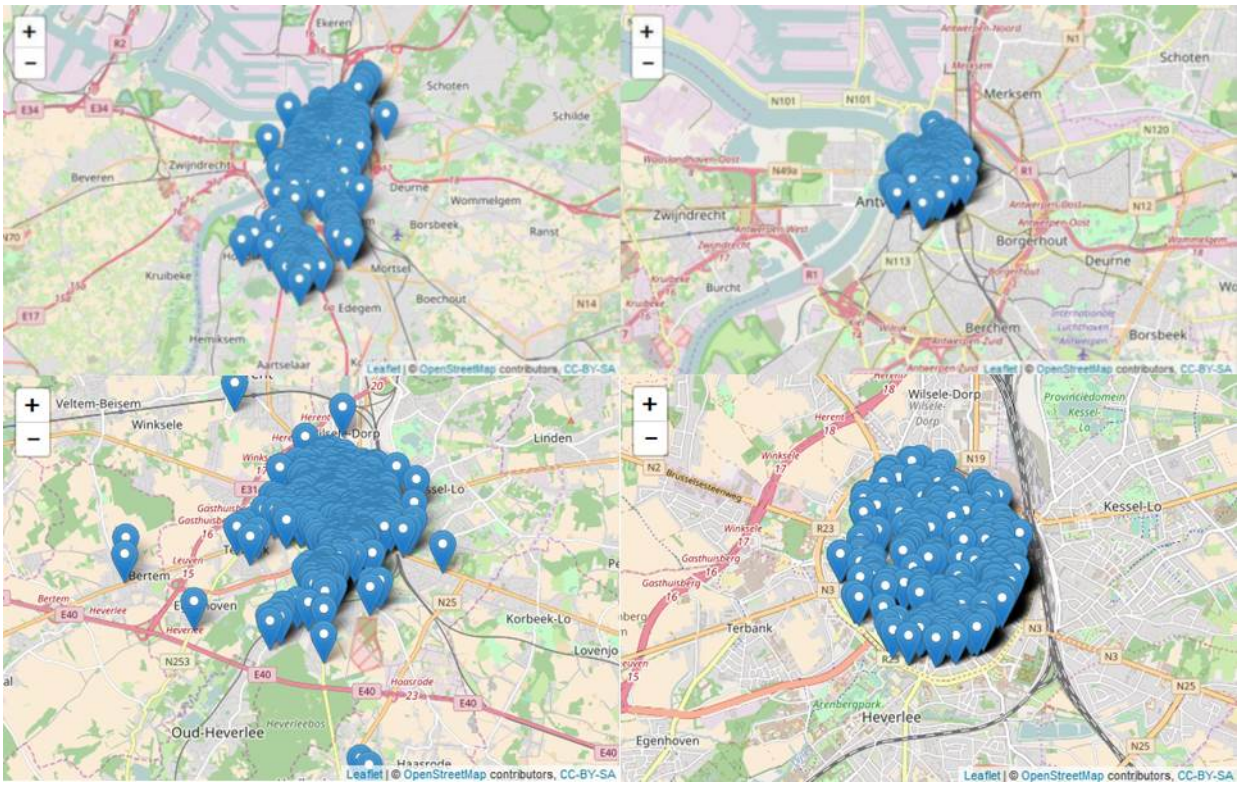
Het huren van studentenkamers is momenteel nog niet opgenomen in de indexberekening. Voorlopig bestaat het huursegment enkel uit private en sociale huurwoningen. Omdat de uitgaven voor studentenkamers toch significant blijken, is het interessant een onderzoek voor dit segment uit te voeren.

Voor studentenkamers is het, in tegenstelling tot private huur, niet mogelijk om een administratieve database van geregistreerde huurcontracten te raadplegen. Dergelijke huurcontracten worden niet vaak geregistreerd en het is in de database moeilijk om te herkennen of een contract al dan niet afgesloten is voor het verhuur van een studentenkamer. Bij de prijsopnames van private huur wordt een steekproef getrokken uit de administratieve database van huurcontracten en worden de huurders gecontacteerd. Voor sociale huur worden sociale verhuurmaatschappijen gecontacteerd.

Via universiteiten zouden contactgegevens kunnen bekomen worden van studenten die een kot huren. Er wordt echter een lage response rate verwacht., Een ander probleem is de korte huurtermijn van studentenkamers. De private huurenquête werkt via een panelsysteem waarbij dezelfde huurwoning doorheen de tijd gevolgd wordt via bevraging van de huurder (ondervraging van de huurder is enkel opgenomen in het desbetreffende K.B.). Dit kan bij studentenkamers een probleem vormen. Gewone private huurcontracten kenmerken zich in principe door een lange duurtijd (3 jaar met telkens een verlenging van 3 jaar). Studentenkamers worden echter gehuurd voor 1 jaar (of soms 10 maanden). Om eenzelfde methodologie als voor de private huurenquête mogelijk te maken, zou dan ook telkens de nieuwe student bereid moeten zijn om deel te nemen aan de enquête. Door deze korte duurtijd en door het stopzetten en afronden van studies zijn er elk jaar opnieuw een hoog aantal kamers te huur. Dit treedt telkens op in een beperkt aantal maanden (juni – september). Al die redenen maken het interessant om de dataverzameling voor dit segment uit te voeren via webscraping via sites waar studentenkamers worden aangeboden.

De volgende gegevens van studentenkamers werden gescrapet voor verschillende studentensteden: prijzen, kamergrootte en adres. Daarna werd ook nog via geocoding een filter toegepast op het adres om het effect van de afstand van de huurlocatie tot de campus te minimaliseren.

Een voorbeeld van geocoding is weergegeven in volgende figuur, waarbij rechts het resultaat staat na filtering op afstand tot de campus:




**Figuur 17: Voorbeeld van geocoding voor 2 studentensteden (Antwerpen en Leuven)**

De index werd berekend voor Antwerpen en Leuven, waarbij de prijzen van juni 2017 vergeleken werden met juli 2016. Beide steden leverden gelijkaardige resultaten:

| Periode | Stad 1 | Stad 2 |
|---|---|---|
| 07-2016 | 100 | 100 |
| 06-2017 | 102,13 | 102,33 |

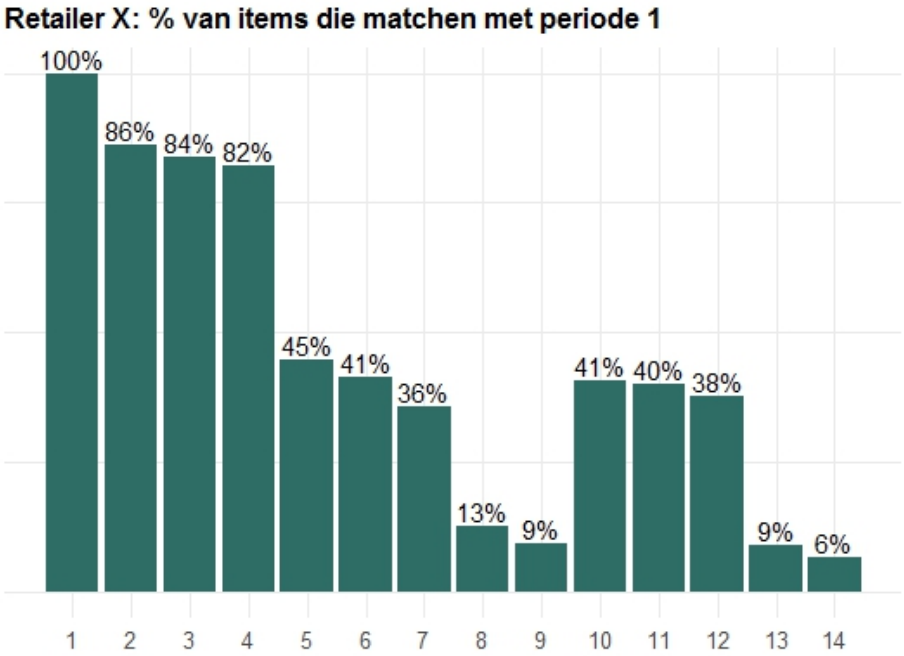
Aangezien de periode van gegevensverzameling vrij kort is (slechts enkele maanden per jaar) blijft het moeilijk om de indexberekening van studentenkamers frequent te testen. Ondertussen werden de scripts wel uitgebreid met extra studentensteden. Dit jaar staat een verdere evaluatie gepland.

# 9. CASE STUDIE – SCHOENEN

De websites van de grootste schoenenwinkels in België worden gescrapet. Het gaat om bedrijven met zowel fysieke winkels als een onlinewinkel. Meerdere keren per week is er een bulkwebscraping. Dat wil zeggen dat alle producten op de website worden gescrapet en dat er geen productselectie (of beperking) gebeurt op voorhand. Data cleaning en -selectie gebeuren tijdens de analysefase.

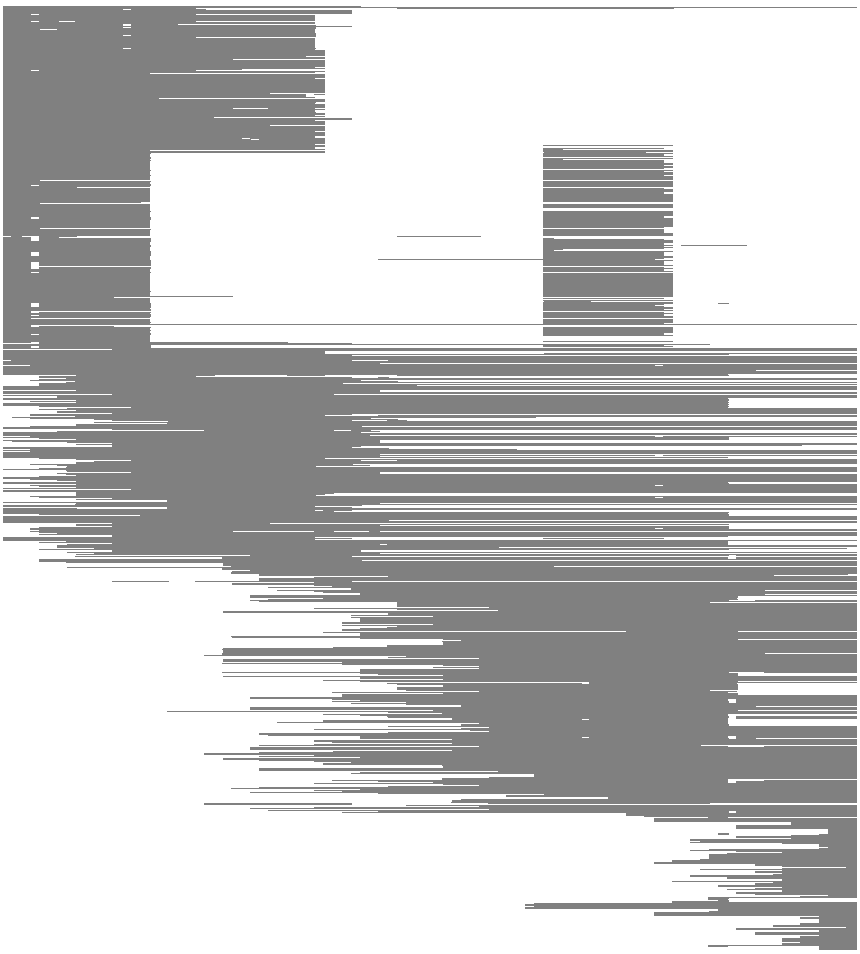
Het bulkscrapen verschilt van de manier waarop de scraping voor consumentenelektronica gebeurde, waarbij specifieke producten geselecteerd werden. De beperking gebeurde daar voornamelijk om alle kenmerken te kunn-en scrapen en omdat deze producten gestandaardiseerd zijn over verschillende websites. Voor schoenen zijn de kenmerken echter beperkt en is er dan ook geen voorafgaande beperking noodzakelijk. Bovendien varieert het aanbod van schoentypes van site tot site, wat het moeilijk maakt om op voorhand een beperking in te stellen.

Producten zoals schoenen (en ook kleding) kennen, nog meer dan consumentenelektronica, een hoge 'attrition rate' (verloop van items). Dat wil zeggen dat er frequent producten in het gamma komen en verdwijnen (bijvoorbeeld omwille van wijzigingen in bepaalde modetrends of ten gevolge van een seizoenswissel). In onderstaande grafiek staat het aantal items dat met periode 1 gematched kan worden voor 1 retailer. Het aantal items dat overeenkomt daalt na 4 maanden al naar minder dan de helft. Het daalt vervolgens naar 9% in maand 9 om dan weer te stijgen in maand 10 tot 12 (één jaar later, dus ongeveer hetzelfde seizoen), waarna het aantal items dat matched weer daalt tot 6%.



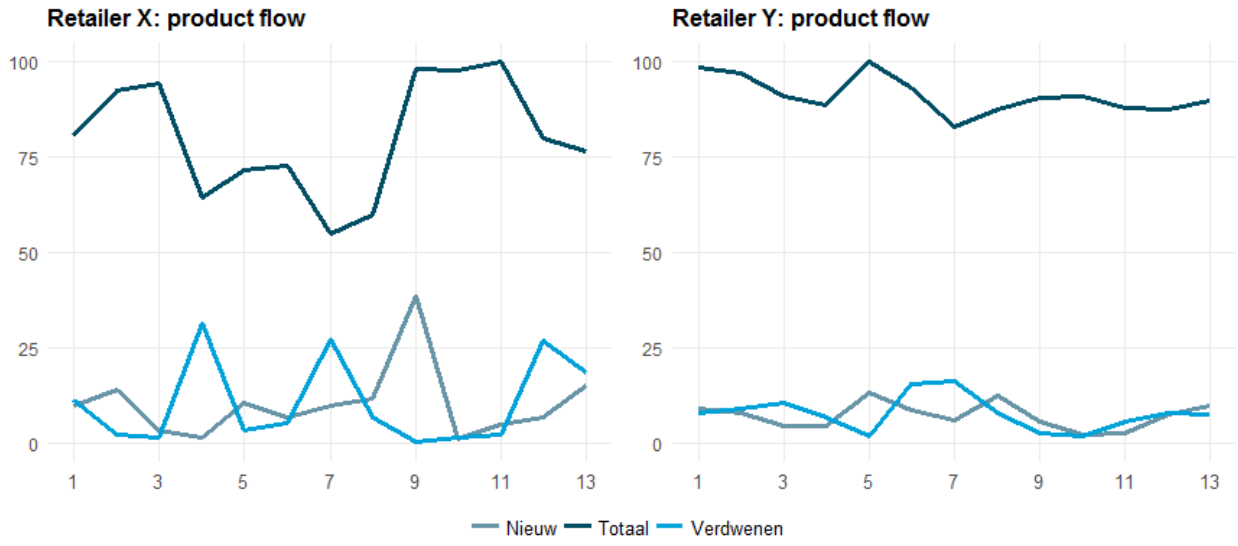

**Figuur 18: Attrition rate van schoenen**

De dynamiek van de gescrapete gegevens kan ook gevisualiseerd worden. In onderstaand schema wordt de beschikbaarheid van een product weergegeven. De items kan men terugvinden op de verticale as. Op de horizontale as wordt de tijd weergegeven. Elke grijze pixel geeft weer wanneer een product beschikbaar is op de website. Verschillende clusters kunnen worden geobserveerd (wijzigingen in productgamma, seizoenspatronen, … ). Het aantal producten dat doorheen de hele periode beschikbaar is, gekenmerkt door een grijze lijn, is zeer beperkt.



Figuur 19: Visualisatie van de beschikbaarheid van schoenen

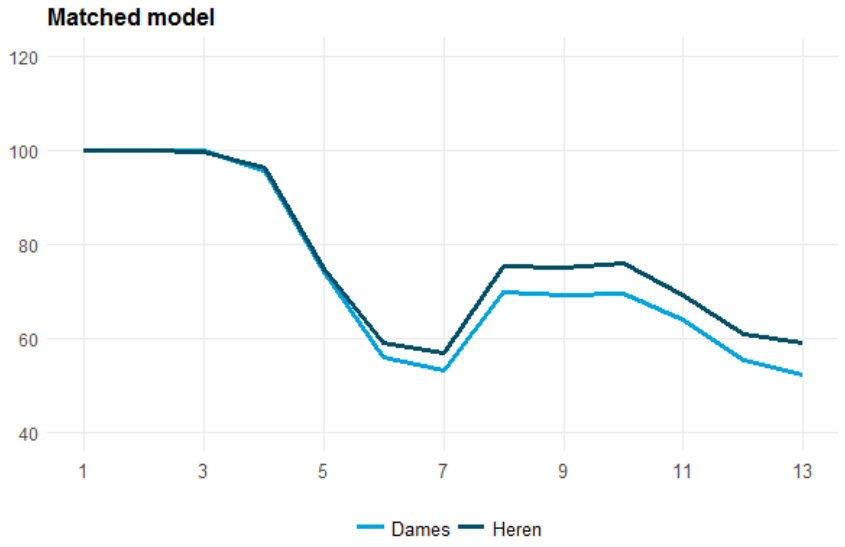
De onderstaande grafiek illustreert voor twee retrailers het totale verloop van producten. Dit totaal wordt geplaatst tegenover het maximaal aantal beschikbare producten tijdens de hele periode (100), samen met het aantal nieuwe en verdwenen producten per periode ten opzichte van het maximaal beschikbaar aantal producten. Zo kan men vaststellen dat voor retailer X het totaal aangeboden producten op de site sterk varieert van periode tot periode, terwijl dit voor retailer Y meer stabiel is.




**Figuur 20: Aantal producten (Totaal, nieuw en verdwenen) bij twee retailers**

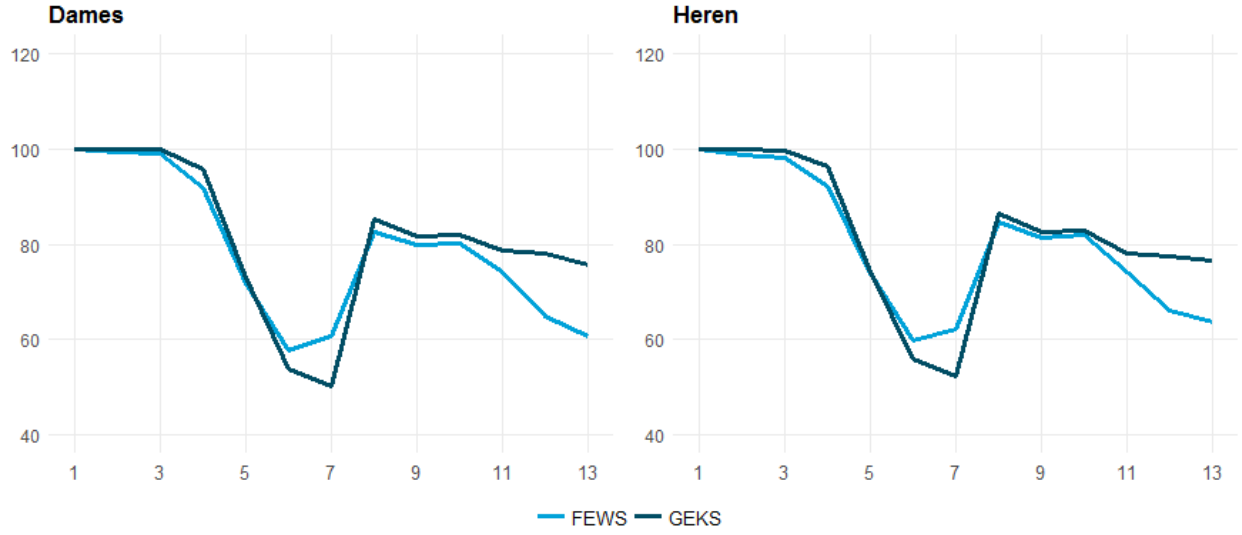
Naast dit dynamische karakter is er nog een ander kenmerk van deze sector: wanneer producten uit het gamma verdwijnen, doen ze dat meestal aan een significant lagere prijs dan die waarmee ze op de markt zijn gekomen.

Indien de indexberekening zou gebeuren op basis van een matched model benadering, waarbij voor items die in consecutieve perioden beschikbaar zijn de prijzen aan elkaar worden gekoppeld, dan wordt een index verkregen die gekenmerkt wordt door een neerwaartse drift, zoals wordt weergegeven in onderstaande grafiek voor retailer X. De neerwaartse drift is nog meer uitgesproken dan bij consumentenelektronica werd vastgesteld. Er is een daling voor zowel heren- als damesschoenen in periode 6 en 7 omwille van solden. Veel van deze producten verdwijnen uit het gamma na deze solden (zie product flow grafiek hierboven) waardoor de index niet terugkeert naar een hoger niveau zoals bijvoorbeeld het niveau in periode 3.




**Figuur 21: Neerwaartse drift van het matched model**

Dit probleem, waarbij items het productgamma verlaten tegen sterk gereduceerde prijzen waarna ze vervolgens niet terugkomen kan ook hier niet opgelost worden door te werken met een multilaterale methode zoals een GEKS-index (in dit geval een GEKS-J) of een FEWS-index. Beide indexen die hieronder worden weergegeven hebben een gelijkaardige evolutie als de matched model Jevons-index.



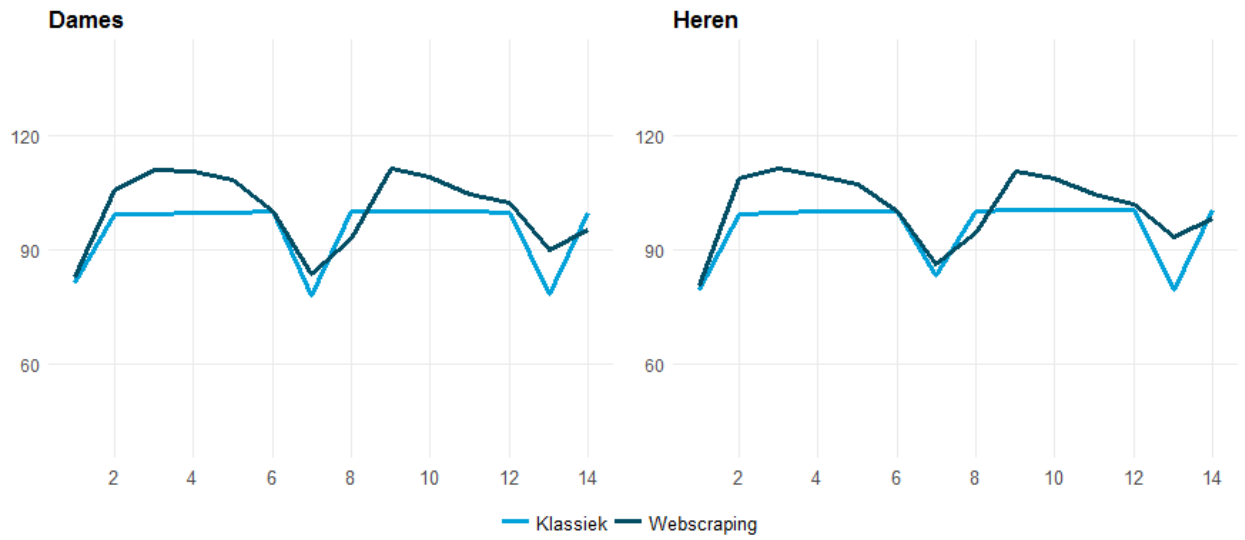

**Figuur 22: Neerwaartse drift bij FEWS en GEKS**

Om de neerwaartse drift te vermijden werd een non-matched model benadering gehanteerd, omdat onvoldoende karakteristieken beschikbaar zijn om een relevante hedonische correctie mogelijk te maken. Hierbij wordt per website eerst een stratificatie gemaakt voor vrouwen- en mannenschoenen, waarna elk segment verder werd opgesplitst in schoentype. Dat gebeurde met behulp van de classificatie die voorzien is op de website van de retailers, aangezien het aanbod kan verschillen van site tot site. Een eenvoudige Jevons-index werd berekend op dit laagste niveau over alle items. Het verschil tussen de wijze waarop de index voor vrouwen- en mannenschoenen (beide een COICOP 5 groep) wordt berekend via webscraping en de klassieke wijze met prijsopnames in winkels (huidige officiële index) wordt weergegeven in volgend schema:

| Klassieke prijsopnames (officiële index) | Webscraping index |
|---|---|
| COICOP 5 | COICOP 5 |
| ↓ | ↓ |
| Steekproef van consumptiesegmenten | Retailer |
| ↓ | ↓ |
| Jevons index | Niet-geharmoniseerde consumptiesegmenten |
| | ↓ |
| | Jevons index |

In de klassieke benadering wordt een meer bottom-up benadering gehanteerd, waar voor een bepaald schoentype - aan de hand van een definitie - prijzen worden opgenomen en de resulterende index wordt berekend op basis van de Jevons-index. Bij webscraping wordt er meer een top-down benadering gebruikt waarbij vertrokken wordt van alle beschikbare schoenen die vervolgens per retailer geaggregeerd worden tot een index voor mannen- en vrouwenschoenen, waarna een globale index berekend kan worden.

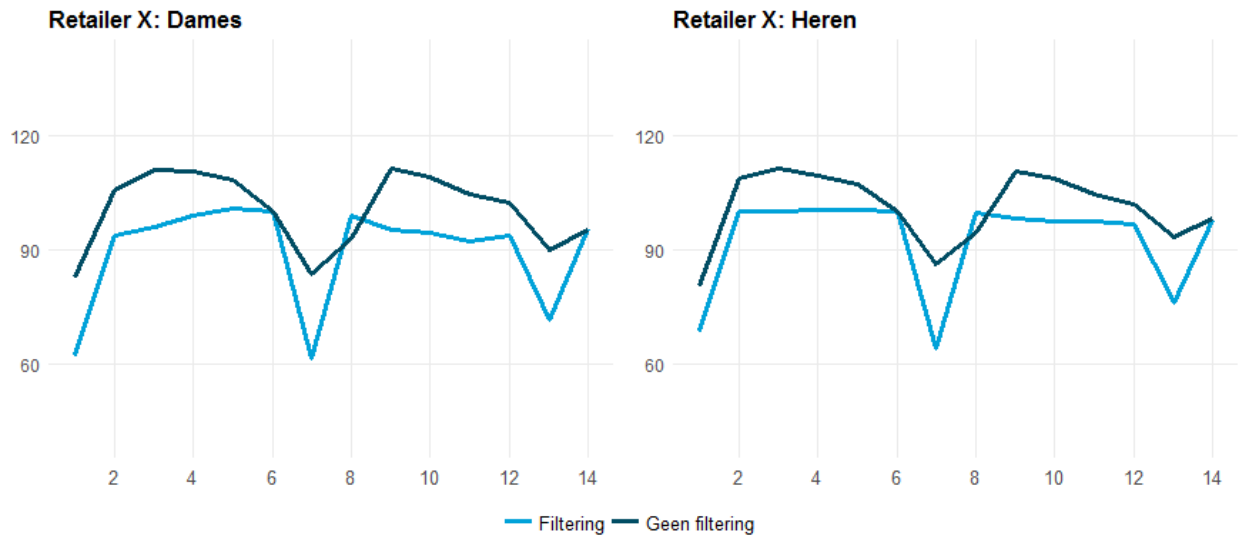
Wanneer de webscrapingindex vergeleken wordt met de officiële indexcijfers, kan een gelijkaardige trend vastgesteld worden. Er is echter een verschil in indexniveau gedurende bepaalde maanden.




**Figuur 23: Vergelijking index tussen klassieke methode en webscraping**

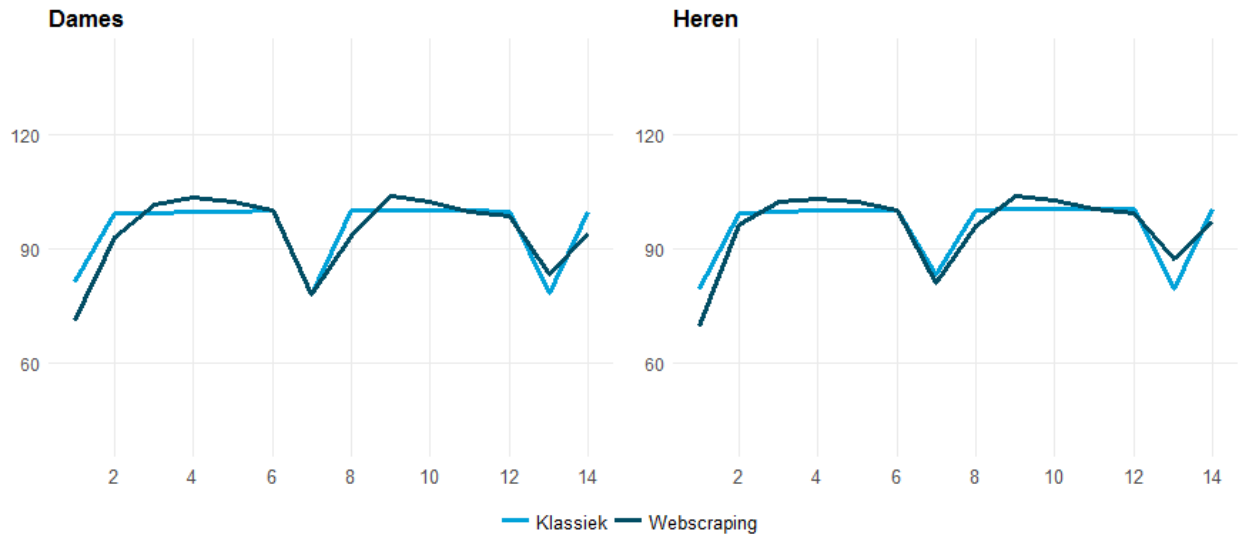
Dat verschil kan verklaard worden door het feit dat in de klassieke prijsopnames de solden enkel worden opgenomen in januari en juli.

Die werkwijze wordt gehanteerd omdat deze solden zeker elk jaar terugkomen en ook een hele maand duren. Dat maakt de planning van prijsopnames gemakkelijker. Voor kortingen die slechts zeer tijdelijk duren en niet voor iedereen beschikbaar zijn, worden geen prijzen genoteerd. In de webscrapingdata zitten deze kortingen wel. Nu valt er een kortingsactie voor retailer X in maand 6 (dat is ook de maand waarbij de index aan 100 is gesteld omdat het ook de basismaand is voor de klassieke prijzen), waardoor het indexniveau in de andere maanden omhoog wordt gebracht omdat vergeleken wordt met dit lage prijsniveau. Indien we deze korting uitzuiveren voor retailer X dan kan men vaststellen dat de index naar beneden schuift.




**Figuur 24: Indexberekening met en zonder filtering van kortingen**

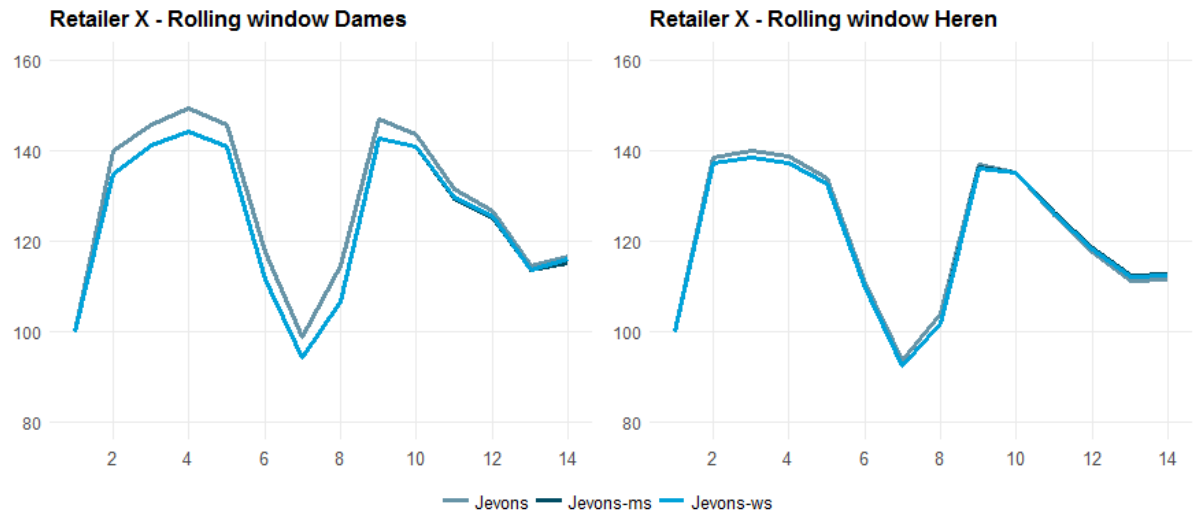
Indien we nu deze index integreren met de indexcijfers van de andere websites dan worden vrij gelijkaardige indexcijfers bekomen voor webscraping en klassieke prijsopnames.




**Figuur 25: Vergelijking index tussen klassieke methode en webscraping met filtering**

In de berekening hierboven werden enkel 'types van schoenen' of consumptiesegmenten gebruikt die gedurende de hele periode beschikbaar zijn.

Omdat bepaalde types van schoenen niet meer worden aangeboden gedurende de hele tijdreeks, gaan daarmee observaties verloren naarmate het aantal perioden waarvoor geobserveerd wordt, stijgt. Bovendien kan een nieuw type van schoen op de markt komen en zou deze dan nooit geïntegreerd worden in het indexcijfer. Het is dus aangewezen om ook hier te werken met rolling window. Aangezien er slechts gegevens zijn voor een periode van 14 maanden, zou een simulatie met de standaardramingsperiode van 13 maanden weinig zinvolle resultaten geven, omdat dan slechts een verschil zou kunnen opgemerkt worden in de 14$^{de}$ maand. Om die reden werd een simulatie gedaan met een kortere vensterperiode, namelijk 8 maanden.




**Figuur 26: Vergelijking tussen standaard Jevons-index, Jevons-index met Movement Splice en Jevons-index met Window Splice (beide met een rolling window van 8 maanden)**

Bovenstaande grafiek geeft de vergelijking tussen een movement splice (Jevons-ms) en een window splice (Jevons-ws), ten opzichte van de hierboven uitgelegde standaard Jevons-index. Het verschil is minimaal, en het verschil tussen beide splicing opties is bijna niet zichtbaar. In praktijk dient de vensterperiode uiteraard langer te zijn dan 8 maanden. Bij een korte vensterperiode bestaat het risico om bepaalde schoentypes in de berekening op te nemen die slechts kort beschikbaar zijn, waardoor telkens het verschil tussen lanceringsprijs en eindprijs in rekening gebracht zou worden (met neerwaartse drift tot gevolg omdat er geen 'bounce back' gebeurt).

De window splice is waarschijnlijk ook te verkiezen boven de movement splice. Bij de window splice methode wordt namelijk voor de 'nieuwe' productgroepen, die voldoende maanden beschikbaar zijn en voor het eerst opgenomen worden in de berekening, de volledige evolutie gedurende die periode in rekening gebracht. Bij de movement splice methode daarentegen heeft enkel de laatste periode-op-periode-beweging een invloed.

Op het laagste niveau wordt er dus een Jevons-index gebruikt; dit is een ongewogen index. Uiteraard kan er online via webscraping geen omzetinformatie teruggevonden worden. Op basis van een aantal karakteristieken kunnen wel gelijkaardige items geïdentificeerd worden. Chessa en Griffioen (2017)[14] kwamen tot volgende conclusie: indien het aantal items als proxy genomen wordt voor het aantal verkochte stuks en vervolgens de omzet berekend wordt door dit te vermenigvuldigen met de gemiddelde prijs voor deze items, worden resultaten bekomen die gelijkaardig zijn aan een index die berekend wordt met scannerdata van dezelfde website. Hun oefening was echter gebaseerd op slechts één website en dus is het moeilijk te veralgemenen.

We hebben getracht om voor één retailer dezelfde oefening te maken en een index te berekenen die rekening houdt met aantal 'verkochte stuks' (met dezelfde proxy als hierboven werd beschreven) en daarna de resulterende index af te zetten tegen een ongewogen Jevons-index. De methode die gebruikt werd om rekening te houden met het aantal 'verkochte stuks' is de 'Quality adjusted unit value - Geary-Khamis methode' (QU-GK).

De QU-GK index (Chessa 2016)[15] wordt gedefinieerd als volgt:

$$P_{QU}^{0,t} = \frac{\sum_{i \in G_t} p_i^t \, q_i^t / \sum_{i \in G_0} p_i^0 q_i^0}{\sum_{i \in G_t} v_i q_i^t / \sum_{i \in G_0} v_i q_i^0} \qquad \textit{Vgl. 39}$$

Waarbij $p$ de prijs is en $q$ het aantal verkochte stuks. De teller geeft dus de evolutie van de totale waarde van alle observaties in een bepaald consumptiesegment. De noemer geeft de evolutie tussen periode 0 en periode $t$ voor een gewogen aantal transacties van elk item in consumptiesegment $i$. De index is dus eigenlijk gelijk aan een waarde index die door een

[14] Chessa, T., Griffioen, R. (2017), *Weights for web scraped data, Comparing scanner data and web scraped data*, CBS (nog niet gepubliceerd)
[15] Chessa, T. (2016), *A new methodology for processing scanner data in the Dutch CPI*, Eurostat Review on National Accounts and Macroeconomic Indicators, 49-69.

hoeveelheidsindex gedeeld wordt, waarbij die laatste gecorrigeerd wordt voor 'kwaliteitsverschillen' tussen producten. De berekening van $v_i$ is niet evident:

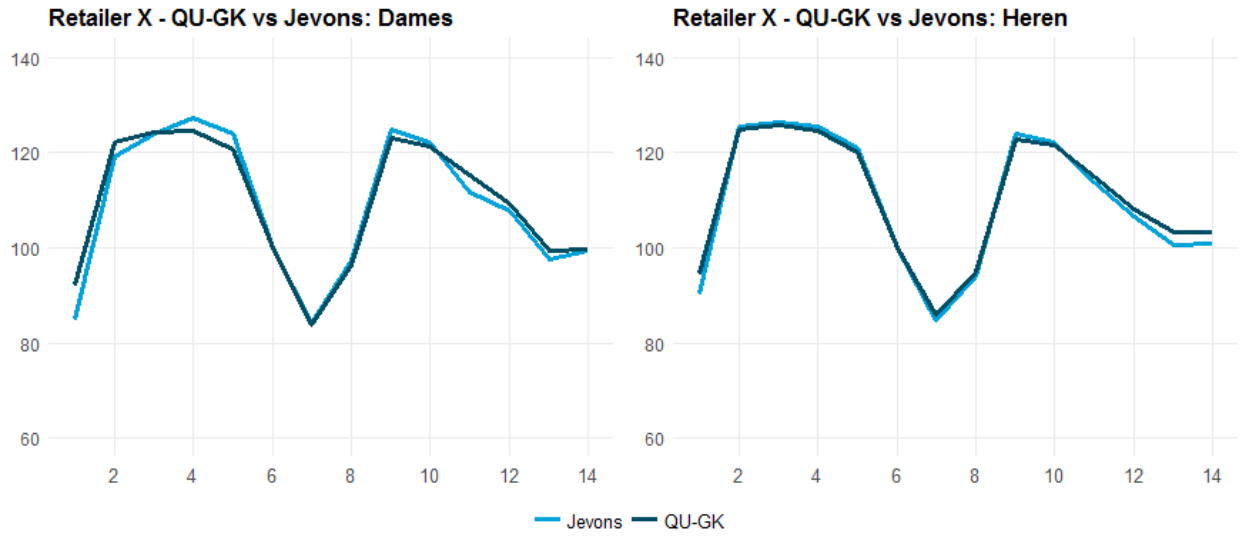
$$v_i = \sum_{z \in T} \varphi_i^z \frac{p_i^z}{P_{QU}^{0,z}} \qquad \textit{Vgl. 40}$$

en

$$\varphi_i^z = \frac{q_i^z}{\sum_{s \in T} q_i^s} \qquad \textit{Vgl. 41}$$

$P_{QU}$ is dus één van de factoren die $v_i$ bepaalt, terwijl $v_i$ ook zelf $P_{QU}$ mee helpt bepalen. Beide vergelijkingen moeten dus op hetzelfde moment berekend worden. Via een iteratieve procedure kan er convergentie optreden. Die wordt bekomen door als startbasis een fictieve index te nemen (bijvoorbeeld 1).

In onderstaande grafiek wordt de resulterende QU-GK index vergeleken met de Jevons-index. Het verschil tussen beide is verwaarloosbaar.




**Figuur 27: Vergelijking Jevons index en QU-GK-index**

Er kruipt echter vrij veel tijd in het definiëren van gelijkaardige items. Bovendien is de veronderstelling om het aantal gelijkaardige items als proxy van het aantal verkochte items te beschouwen niet altijd correct. Daarom is het misschien pragmatischer om te werken met een ongewogen Jevons-index.

Ondertussen werd nog meer data verzameld via webscraping en wordt deze case studie verder onderzocht.

# 10. MACHINE LEARNING

Aangezien bij webscraping heel veel data wordt opgehaald, is het mogelijk dat er ook data aanwezig is die minder interessant is voor verdere verwerking of berekeningen. Met behulp van machine learning en algoritmes kan de opgehaalde data gefilterd en geklasseerd worden, zodat enkel de gegevens waarin interesse is, overblijven. Het is ook nodig om de verzamelde data in algemene categorieën te kunnen klasseren om zo de indices per categorie te kunnen berekenen.

Een eenvoudig voorbeeld is de scraping van dvd's: van bepaalde websites worden een aantal gegevens (titel, beschrijving, prijs,…) gescrapet voor de dvd's die beschikbaar zijn. Voor indexberekening is het echter mogelijk dat er enkel interesse is in 'single' dvd's (homogene producten), dus niet in dvd-boxen, speciale edities,… . Hier kan dan een klasseringsalgoritme (machine learning) op toegepast worden om enkel de 'interessante' gegevens over te houden. Er wordt dan gekeken naar de productomschrijving van de dvd's. Op basis van de woorden die voorkomen in de omschrijving kan er beslist worden tot welke categorie de dvd behoort (tekst-classificatie). Die classificatie zou natuurlijk ook in theorie volledig manueel geprogrammeerd kunnen worden. Bijvoorbeeld door het bijhouden van een aantal 'woordenboeken', wanneer een woord (of x-aantal woorden) teruggevonden worden in de voorgeprogrammeerde woordenboeken, dan zou een item toegewezen worden aan een bepaalde categorie op basis van beslissingsregels. Dat vergt uiteraard veel manueel opzoekwerk. Elke wijziging van de woordenboeken moet bovendien opnieuw gecontroleerd worden op consistentie met wat voordien geklasseerd werd. Per nieuwe observatie moeten de woordenboeken en beslissingsregels ook weer aangepast worden. Dat vergt allemaal vrij veel tijd. Via machine learning kan dit alles grotendeels geautomatiseerd worden.

Om machine learning te kunnen toepassen, moet er beslist worden welke kenmerken het model moet gebruiken om de classificatie uit te voeren. Om items te klasseren, moeten nieuwe items vergeleken worden met items die reeds geklasseerd zijn. De gegevens moeten zodanig verwerkt worden dat items met elkaar kunnen vergeleken worden. Een manier om dat te doen, is alle karakteristieken om te zetten in numerieke waarden. Voor tekstclassificatie wordt hiervoor vaak gewerkt met de woordfrequentie in de omschrijvingen van de data. De 'numerieke waarde' van een woord komt dan overeen met de frequentie van het woord. Er zijn verschillende algoritmes binnen machine learning beschikbaar die tekstclassificatie kunnen uitvoeren: o.a. *K-nearest neighbours* (KNN), *naive bayes classifier*, *random forest* en *support vector machine*. Elk algoritme gebruikt een andere manier om items met elkaar te vergelijken. We beschrijven deze algoritmes verder in het detail.

Algemeen wordt in eerste instantie een trainingsdataset opgesteld, waarbij de gegevens manueel geklasseerd worden. Aan de hand van die dataset kan elk algoritme een model bouwen om nieuwe data te klasseren op basis van de gelijkenissen tussen de trainingsdataset en de nieuwe data. Daarna kan elk algoritme getest worden op een set van testdata, waarna kan beslist worden welk algoritme het beste resultaat geeft.

Voordat het algoritme wordt toegepast op nieuwe data, wordt deze data ook nog 'gecleaned'. Dat houdt in dat de data bepaalde transformaties ondergaan, zodat ze beter bruikbaar zijn in de verdere verwerking en analyse. Het is ook nodig om deze transformaties uit te voeren zodat alle gegevens op een gelijke manier behandeld worden. Hierdoor kan een nieuw item steeds op dezelfde manier vergeleken worden met de items in de trainingsdata. Door de transformaties worden lange omschrijvingen omgezet in de belangrijkste woorden van de omschrijving. Voorbeelden van *cleaning* zijn:

- Omzetting naar kleine letters: om ervoor te zorgen dat alle omschrijvingen op dezelfde manier gelezen worden, is het handig om alles te transformeren naar kleine letters.
- Stopwoorden verwijderen: stopwoorden zijn woorden die vaak gebruikt worden, maar nooit veel betekenis hebben (bv. lidwoorden, voegwoorden, …). Deze woorden bieden geen meerwaarde aan de klassering en worden dus vaak weggelaten om de werking van het model te verbeteren.
- Verwijderen van gemeenschappelijke woorden: gemeenschappelijke woorden zijn onbelangrijk bij het klasseren in verschillende categorieën. Bijvoorbeeld bij het scrapen van dvd's is het woord 'dvd' in de omschrijving op zich onbelangrijk aangezien alles wat gescrapet wordt een dvd is.
- Verwijderen van leestekens en spaties: deze geven geen meerwaarde in de klassering.
- Het terugbrengen van woorden naar de stam van de woord (*stemming*): bijvoorbeeld voor 'seizoenen' is de stam 'seizoen'.

Na de toepassing van het algoritme kan de nieuw geklasseerde data toegevoegd worden aan de trainingsdata waardoor het model steeds intelligenter en betrouwbaarder wordt.

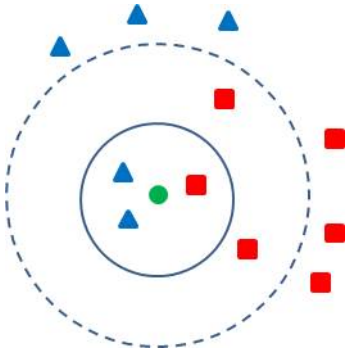
## 10.1. KNN

KNN staat voor K-nearest-neighbours-method. De KNN-methode vertrekt van het principe om voor alle nieuwe items te kijken naar de k-dichtstbijzijnde (trainings)buren in de trainingsdata. Het nieuwe item krijgt dan dezelfde classificatie als de meerderheid van de 'k-nearest neighbours'. Als k=1, wordt er voor het nieuwe item gekeken naar het dichtstbijzijnde item uit de trainingsdataset en de classificatie van deze 'buur' wordt dan ook toegewezen aan het nieuwe item.

De keuze van k wordt iteratief bepaald en op basis van de accuraatheid wordt de optimale k gekozen en toegepast. Als er slechts twee mogelijke classificaties zijn, is het logisch dat k oneven moet zijn om 'een gelijkspel' uit te sluiten. Indien bij hogere waarden van k een gelijkspel voorkomt, kiest het algoritme arbitrair welke categorie er wordt toegewezen.

Een 'dichtstbijzijnde buur' verwijst naar een afstandsbetekenis. Er moet dus een afstand bepaald worden tussen het nieuwe item en de items uit de trainingsdata. Alle karakteristieken van een item kunnen gezien worden als verschillende dimensies in de ruimte. De waarde van de karakteristieken zorgen dan voor de plaats in de ruimte waar het item zich bevindt (de coördinaten). Wat betreft tekstclassificatie stellen alle woorden (=karakteristieken) hier een bepaalde ruimtelijke dimensie voor en een nieuw item komt dan overeen met de coördinaten van alle woorden van de omschrijving in de ruimte. Zo hebben alle items een bepaalde plaats in de ruimte en kan de vergelijking tussen items gemaakt worden op basis van de (Euclidische) afstand tussen de items. De classificatie die het meeste voorkomt binnen de k-dichtstbijzijnde buren wordt dan toegewezen aan het nieuwe item.

Nieuwe ongekende data, die geen overeenkomstige karakteristieken hebben binnen de trainingsset, worden 'hernoemd' tot de defaultwaarde binnen de trainingsdata.



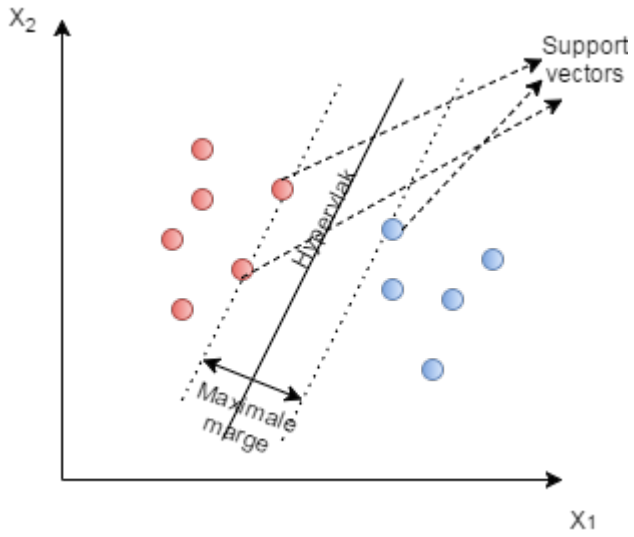
**Figuur 28: Visualisatie van KNN-methode**

Bovenstaande figuur geeft een voorbeeld van KNN-classificatie van items met 2 karakteristieken (lineair, x- en y-coördinaat). De bedoeling is om het nieuwe item (groene stip) te klasseren in ofwel de groep van rode vierkanten ofwel de groep van blauwe driehoeken. Als k=3 wordt er gekeken naar de 3 dichtstbijzijnde objecten (binnenste cirkel), op basis van dit algoritme zou de cirkel toegewezen worden aan de groep van blauwe driehoeken. Wanneer k=5 (buitenste cirkel) heeft de meerderheid van de 'buren' als classificatie 'rood vierkant', waardoor de cirkel in deze categorie zal worden ondergebracht.

## 10.2. Support vector machine

De support vector machine-methode werkt, net zoals de KNN-methode, met ruimtecoördinaten. Het SVM-algoritme maakt dan theoretisch gezien een 'zo goed mogelijke' scheiding (hypervlak) tussen de verschillende categorieën. Door te kijken aan welke kant van het hypervlak het nieuwe dataobject ligt, kan het SVM-model het nieuwe item in de juiste categorie onderbrengen. Hoe verder een item van het hypervlak ligt, hoe 'meer' het tot die categorie behoort. Er zijn verschillende manieren om de categorieën van elkaar te scheiden. De 'zo goed mogelijke' scheiding houdt in dat de afstand tussen het scheidingsvlak en de dichtsbijgelegen objecten van elke klasse (de marge) zo groot mogelijk is.

Onderstaande grafiek geeft schematisch weer hoe de classificatie van observaties in twee klassen kan gebeuren via lineaire SVM.




**Figuur 29: Voorstelling van classificatie via support vector machine-algoritme**

De punten die op de stippellijnen liggen worden de support vectors genoemd omdat ze het hypervlak ondersteunen. De grootst mogelijke marge wordt bepaald door de evenwijdige hypervlakken met de support vectors. Het support vector machine-algoritme zal, op basis van de omschrijving en de toegewezen classificatie in de trainingsdata, een model creëren dat nieuwe items toewijst aan één van de categorieën.

## 10.3. Naive Bayes classifier

De Naive Bayes classifier gebruikt kanstheorie en de stelling van Bayes om de categorie van nieuwe data te voorspellen. De stelling van Bayes zegt dat:

$$P(x|y) = \frac{P(y|x) * P(x)}{P(y)} \qquad \textit{Vgl. 41}$$

Waarbij: $P(x) =$ de kans dat $x$ zich voordoet en $P(x|y) =$ de (voorwaardelijke) kans dat $x$ zich voordoet gegeven dat $y$ zich voordoet.

$$P(\text{categorie 1}|\text{omschrijving A}) = \frac{P(\text{omschrijving A}|\text{categorie 1}) * P(\text{categorie 1})}{P(\text{omschrijving A})} \qquad \textit{Vgl. 42}$$

Het Naive Bayes classifier algoritme zal voor elk nieuw item de kans berekenen op elke mogelijke categorie en de categorie met de meeste waarschijnlijkheid wordt toegewezen aan het nieuwe item. De kansberekening gebeurt op basis van de omschrijving van het item. De formule wordt dan:

*'Naive'* verwijst naar het feit dat elk woord in de omschrijving onafhankelijk van de andere woorden wordt beschouwd. Anders zou de kans dat een bepaalde omschrijving tot een categorie behoort vaak 0 zijn, tenzij net exact dezelfde omschrijving reeds in de trainingsdata zit. Door assumptie van onafhankelijkheid is de kans van een omschrijving gelijk aan het product van alle kansen van alle woorden uit de omschrijving. Deze individuele woorden zullen met een grotere waarschijnlijkheid in de trainingsdata voorkomen, waardoor de kans verschillend zal zijn van 0. Wiskundig wordt de onafhankelijkheid genoteerd als:

$$P(\text{omschrijving A}|\text{categorie 1}) = \prod_i P(A_i|\text{categorie 1}), \qquad \textit{Vgl. 43}$$

met $A_i$de verschillende woorden in omschrijving $A$.

Door de onafhankelijkheid is de volgorde van de woorden in een omschrijving niet van belang. Bij Naive Bayes classifier wordt dus steeds gekeken naar de frequentie van een woord in de trainingsdata en de overeenkomstige categorie, er moet dus enkel geteld worden hoeveel keer een bepaald woord voorkomt in een bepaalde categorie.

Door toepassing van deze formules kan de kans berekend worden voor nieuwe data:

$$\begin{aligned} P(\text{categorie 1 }|\text{omschrijving A}) &= \frac{P(\text{omschrijving A }|\text{categorie 1}) * P(\text{categorie 1})}{P(\text{omschrijving A})} \\ &= \frac{\prod_i P(A_i|\text{categorie 1}) * P(\text{categorie 1})}{P(\text{omschrijving A})} \end{aligned} \qquad \textit{Vgl. 44}$$

Dit gebeurt voor elke mogelijke categorie:

$$P(\text{categorie 2 }|\text{omschrijving A}) = \frac{\prod_i P(A_i\ |\text{categorie 2}) * P(\text{categorie 2})}{P(\text{omschrijving A})} \qquad \textit{Vgl. 45}$$

Waarbij omschrijving A bestaat uit de woorden A1, A2, A3,… . De categorie met de hoogste waarschijnlijk wordt uiteindelijk toegewezen aan de nieuwe data.

Wanneer echter een bepaald woord nog niet in de trainingsdataset voorkomt, zal de kans hiervan toch 0 zijn (bv. $P(A_1|cat\ 1) = 0$ ), waardoor er geen resultaat bekomen wordt. Dit probleem kan eventueel door een Laplace-correctie omzeild worden. Een mogelijke correctie is:

$$P(A_i \mid \text{cat } 1) = \frac{\# A_i + 1}{\# w_i \text{ cat } 1 + \# w_i T} \qquad \textit{Vgl. 46}$$

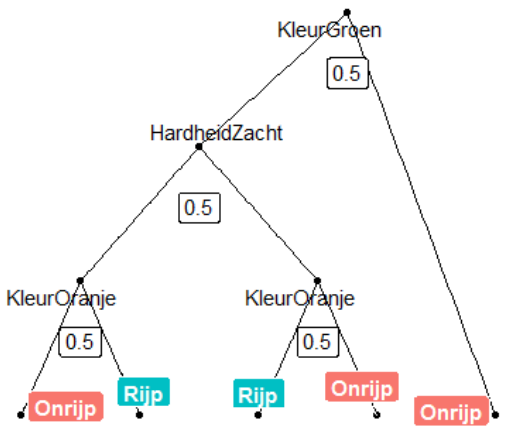
Met $\# A_i$de frequentie van woord $A_i$, $\# w_i$ cat 1 het aantal woorden in categorie 1 en $\# w_i T$ het aantal woorden in de trainingsdataset. Ook is het mogelijk om geen rekening te houden met de 'onbekende' woorden. De kans hiervan wordt dan als onbestaande beschouwd. Doordat de nieuwe data steeds toegevoegd wordt aan de volledige dataset, breidt deze constant uit en kan er met nieuwe woorden in de toekomst wel rekening gehouden worden.

## 10.4. Random Forests

Random Forests bestaan uit een verzameling van beslissingsbomen (decision trees). In het kader van classificatie is het resultaat van zo'n beslissingsboom dan een bepaalde categorie. Elke beslissingsboom voorspelt de categorie aan de hand van bepaalde beslissingsregels. Die regels worden opgesteld op basis van de trainingsdata, waarbij de omschrijving en de categorie gekend zijn. Door de opeenvolging van regels en uitkomsten ontstaat een boomstructuur, waarbij op elk knooppunt een beslissing moet genomen worden. Afhankelijk van de beslissing wordt een bepaald pad genomen richting uitkomst (categorie). De verzameling van alle mogelijke beslissingsbomen is het Random Forest, de uiteindelijke voorspelling van de categorie volgt uit de combinatie van alle resultaten van de beslissingbomen. De classificatie met de meerderheid aan stemmen van de beslissingsbomen wint. Normalerwijze wordt ongeveer een 500-tal beslissingsbomen gecreëerd. Naarmate het aantal beslissingsbomen stijgt, zal de 'out-of-bag-error' (misclassificatie-percentage) dalen. In de praktijk daalt dat percentage vaak al heel sterk na een 100-tal bomen. De waarschijnlijkheid van een correcte classificatie van een waarde kan ook eenvoudig worden berekend. Als 400 bomen iets in categorie A klasseren en 100 in categorie B dan is de geraamde waarschijnlijkheid van een correcte classificatie door het model 80%.

In een Random Forest worden de beslissingsbomen telkens opgebouwd op basis van verschillende samples van de trainingsdata *('bootstrapping with replacement')*. Elke sample, die overeenkomt met 63,2% van de observaties in de volledige trainingstest, wordt gebruikt als trainingsset voor het creëren van een beslissingsboom. De vertakking aan elk knooppunt is de best mogelijke vertakking in een deelverzameling van de kenmerken (i.e. woorden). De grootte van deze deelverzameling van kenmerken wordt stabiel gehouden voor dezelfde beslissingsboom. De uiteindelijke behouden grootte van de deelverzameling op elk knooppunt wordt bepaald door het Random Forest-model meerdere keren uit te voeren met een verschillende grootte en vervolgens deze te selecteren met de laagste out-of-bag-error. Elke beslissingsboom maakt dus uiteindelijk besluiten qua classificatie op basis van verschillende variabelen (knooppunten). De resterende 38,8% observaties uit de volledige trainingsest die niet gebruikt zijn voor het opstellen van de beslissingsboom worden gebruikt om de 'out-of-bag error' te berekenen voor die boom. Aggregatie van alle misclassificatie-percentages over alle bomen geeft het globale misclassificatie-percentage. Een knooppunt stelt een bepaalde karakteristiek voor en de hieruit volgende takken staan voor de verschillende waarden van deze karakteristiek. In tekst-classificatie maakt elk knooppunt gebruik van woorden om te beslissen welke richting gekozen moet worden. Een knooppunt komt dus overeen met een woord. Er zijn dan twee mogelijke richtingen: de observatie bevat het woord of de observatie bevat het woord niet. In het begin van de beslissingsboom staan in principe de woorden die het grootste onderscheid maken. Door de willekeurige deelverzameling is het aantal knooppunten in elke boom ook verschillend.

De volgende figuur geeft een eenvoudig voorbeeld van een beslissingsboom. De dataset waarop dit gebaseerd is, bestaat uit de omschrijving van enkele sinaasappelen (kleur en hardheid) en de categorieën 'Rijp' of 'Onrijp':




**Figuur 30: Voorbeeld van een beslissingsboom**

## 10.5. Case Studie - Kledij

Gedurende 20 maanden werd wekelijks de informatie van een kledingwebsite gescrapet en opgeslagen. Deze gegevens werden opgeladen in de software R, waar ze eerst aan een cleaning werden onderworpen. Zoals eerder vermeld houdt dat in dat de data zodanig getransformeerd wordt dat enkel de bruikbare gegevens overblijven.

De praktische testing werd uitgevoerd voor twee categorieën, hierna segment 1 en segment 2 genoemd. Voor beide werd bepaald welke de meest voorkomede subcategorieën zijn. Deze subcategorieën werden dan uit de volledige data geselecteerd, dit waren voor segment 1 en 2 respectievelijk 5 en 3 subcategorieën.

Een *document term matrix*[16] werd aangemaakt, deze bevat alle verschillende woorden uit de itembeschrijvingen. Met behulp van een filter worden enkel de relevante (frequentie hoger dan 1) woorden behouden. Daarna werd de document term matrix omgevormd tot een dataframe, hierop kunnen de modellen getraind en kan de classificatie getest worden. De data wordt opgesplitst in een trainingsset en een testset (bv. 85% training, 15% test). Beide datasets behouden de verdeling in subcategorieën zoals ze in de ganse dataset bepaald zijn, anders zou men onevenwichtige training- en testdatasets kunnen bekomen.

Op deze 2 datasets werden de vier machine learning-methodes uitgevoerd. Hieronder staan de resultaten qua accuraatheid voor beide segmenten:

| Methode | Segment 1 | Segment 2 |
|---|---|---|
| KNN | 76.05% | 85.28% |
| SVM | 73.82% | 83.30% |
| Naive Bayes | 70.89% | 79.53% |
| Random Forest | 79.62% | 86.13% |

[16] Een document term matrix is een matrix die de frequentie weergeeft van de woorden die voorkomen in een reeks van omschrijvingen. De kolommen stellen de woorden voor, de rijen de omschrijvingen. Wanneer een woord eenmaal in een omschrijving voorkomt, is dit element van de matrix gelijk aan 1, als het niet voorkomt is het element 0.

In beide tests bleek de Random Forest-methode het meest accuraat. Er moet echter opgemerkt worden dat de drie eerste methodes quasi onmiddellijk een resultaat leveren, in tegenstelling tot de Random Forest-methode die een iets langere runtijd (+/- 10 min) heeft.

## 10.6. Case Studie – dvd's & Blu-rays

Voor dit onderzoek werden dagelijks de 100 best verkochte dvd's en Blu-rays gescrapet van een online retailer. De resulterende dataset bevatte naast de gewone 'single' dvd's en Blu-rays ook speciale edities en tv-series. Zoals in het begin van dit hoofdstuk reeds werd vermeld is het voor een indexberekening belangrijk om een meer homogeen product te bekijken (bv. enkel de 'single' dvd's). Om deze selectie automatisch uit te voeren werd gebruik gemaakt van supervised machine learning.

Ongeveer 720 titels werden willekeurig gekozen en geklaseerd in 2 categorieën: behouden of uitsluiten. In eerste instantie werd de verzamelde data gecleaned d.m.v. standaard tekst mining. Witruimtes en leestekens werden verwijderd, alle tekst werd omgezet in kleine letters en bepaalde woorden werden herleid tot de essentie ('stemming'). Daarna werd de dataset opgesplitst in een trainingset en testset (75% - 25%) die de verdeling respecteerde van behouden of uitsluiten in de volledige dataset.

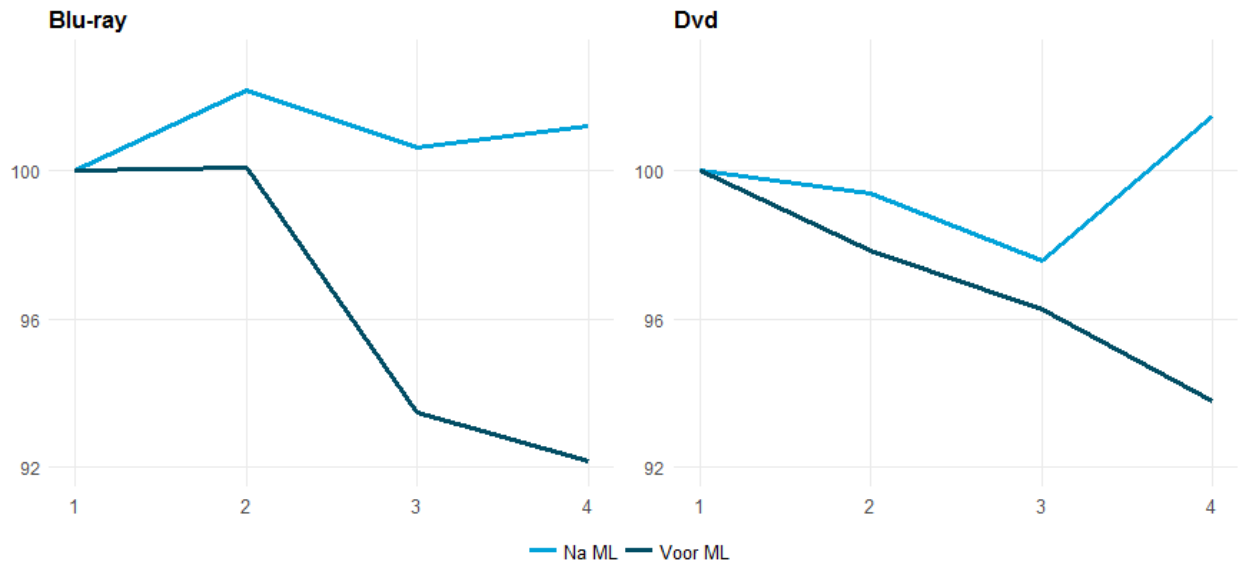
De 4 machine learning-methodes werden toegepast en de resultaten werden vergeleken met de gewenste classificatie.

| Methode | Accuraatheid |
|---|---|
| KNN | 91.95% |
| SVM | 95.87% |
| Naive Bayes | 92.96% |
| Random Forest | 95.75% |

Het beste resultaat wordt hier bekomen met de SVM-methode. Ook Random forest levert een heel hoge accuraatheid, maar opnieuw moet er opgemerkt worden dat de runtijd hiervoor steeds iets hoger ligt dan bij de andere methodes.

De SVM-methode werd eerder al getest op basis van een trainingsset van 300 observaties. Het model werd daarna toegepast op 600 nog niet geklasseerde titels en er werd een Jevons-index berekend. Om een neerwaartse drift (dvd's verlaten de markt aan een lagere prijs dan de prijs waarmee ze op de markt komen) te vermijden werd een unmatched model gebruikt. Dat gaf uiteindelijk de volgende resultaten:

| | Blu-ray | | DVD | |
|---|---|---|---|---|
| Maand | Voor ML | Na ML | Voor ML | Na ML |
| 1 | 100 | 100 | 100 | 100 |
| 2 | 100,08 | 102,16 | 97,85 | 99,38 |
| 3 | 93,48 | 100,65 | 96,27 | 97,59 |
| 4 | 92,16 | 101,19 | 93,80 | 101,47 |




**Figuur 31: Index voor Blu-ray en dvd voor en na toepassing van machine learning**

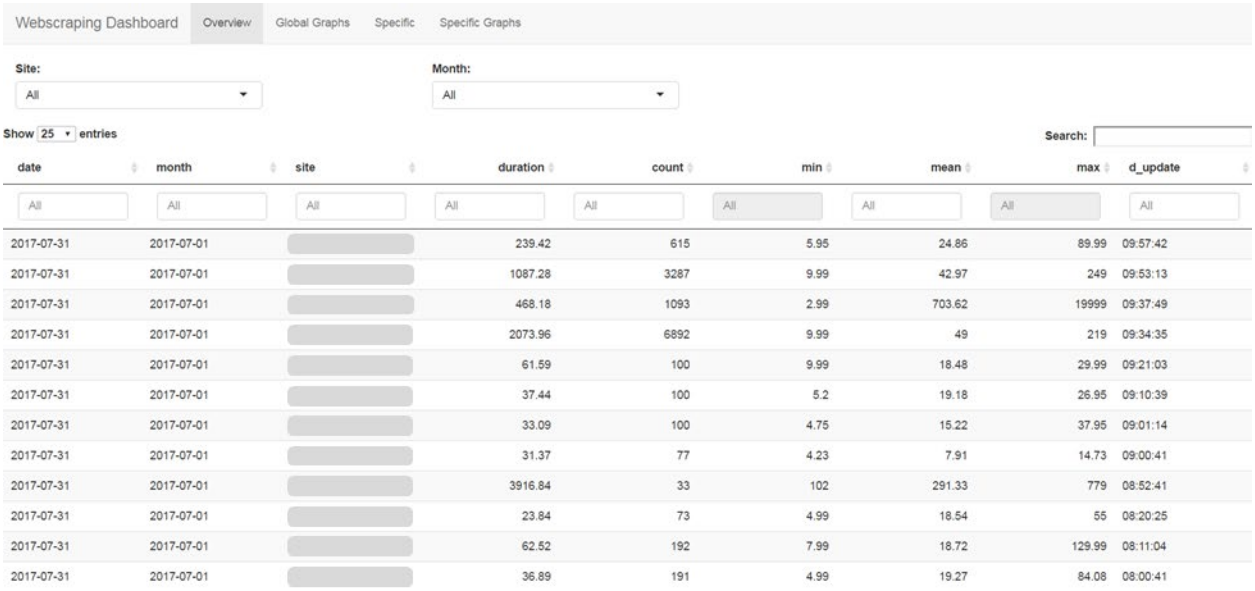
Voor de toepassing van machine learning (ML) merken we een daling in de prijsevolutie. Die daalt doordat in het begin van de prijsopnames ook series en dvd-boxen aanwezig waren in de observaties. Na verloop van tijd verdwenen dergelijke observaties uit het gamma, waardoor de index daalde. Door machine learning toe te passen op de volledige dataset werden series en dvd-boxen a priori al uitgesloten van de indexberekening. Hierdoor is er geen neerwaartse drift aanwezig in de index (Na ML).

# 11. DASHBOARD

Wanneer de scripts uiteindelijk automatisch runnen, is het natuurlijk noodzakelijk om de resultaten te blijven controleren. Daarnaast moeten wijzigingen in de websites ook opgevangen worden zodat de scripts aangepast kunnen worden aan de nieuwe structuur van de websites.

Om de verschillende webscraping-scripts te kunnen monitoren werd een dashboardapplicatie ontwikkeld. Zo kunnen de outputs worden opgevolgd en indien nodig kunnen de scripts aangepast worden. Enerzijds moet er nagegaan worden of het aantal gescrapete records in lijn ligt met de verwachtingen en de vorige resultaten. Daarnaast moeten de resultaten op zich ook gecontroleerd worden. Zo is het essentieel dat er prijzen worden verzameld. Wanneer er dus voor dit kenmerk (prijs) veel ontbrekende waarden zijn, zal de website en het script moeten nagekeken worden.

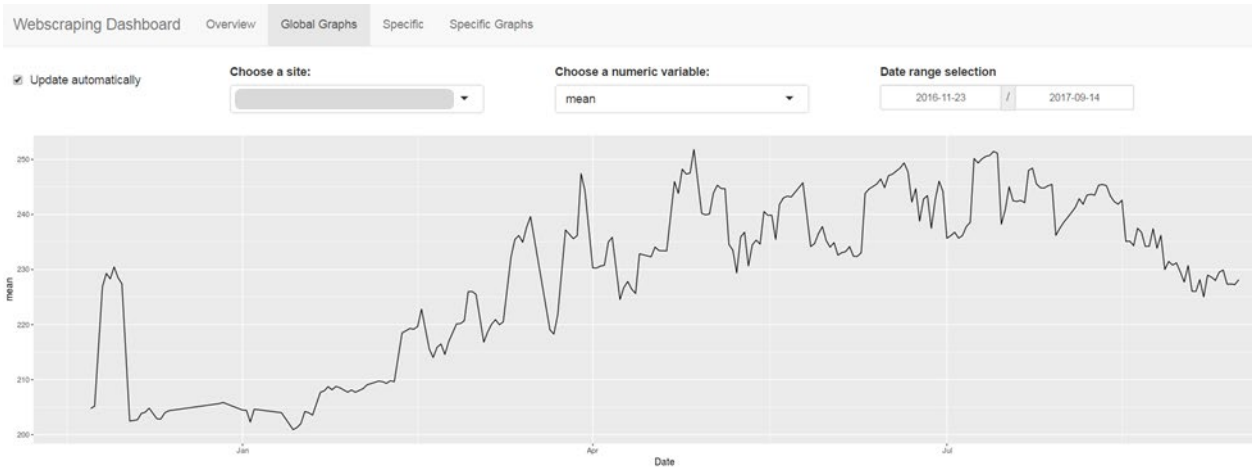
Hieronder staat een printscreen van het webscraping-dashboard:




| date | month | site | duration | count | min | mean | max | d_update |
|---|---|---|---|---|---|---|---|---|
| 2017-07-31 | 2017-07-01 | | 239.42 | 615 | 5.95 | 24.86 | 89.99 | 09:57:42 |
| 2017-07-31 | 2017-07-01 | | 1087.28 | 3287 | 9.99 | 42.97 | 249 | 09:53:13 |
| 2017-07-31 | 2017-07-01 | | 468.18 | 1093 | 2.99 | 703.62 | 19999 | 09:37:49 |
| 2017-07-31 | 2017-07-01 | | 2073.96 | 6892 | 9.99 | 49 | 219 | 09:34:35 |
| 2017-07-31 | 2017-07-01 | | 61.59 | 100 | 9.99 | 18.48 | 29.99 | 09:21:03 |
| 2017-07-31 | 2017-07-01 | | 37.44 | 100 | 5.2 | 19.18 | 26.95 | 09:10:39 |
| 2017-07-31 | 2017-07-01 | | 33.09 | 100 | 4.75 | 15.22 | 37.95 | 09:01:14 |
| 2017-07-31 | 2017-07-01 | | 31.37 | 77 | 4.23 | 7.91 | 14.73 | 09:00:41 |
| 2017-07-31 | 2017-07-01 | | 3916.84 | 33 | 102 | 291.33 | 779 | 08:52:41 |
| 2017-07-31 | 2017-07-01 | | 23.84 | 73 | 4.99 | 18.54 | 55 | 08:20:25 |
| 2017-07-31 | 2017-07-01 | | 62.52 | 192 | 7.99 | 18.72 | 129.99 | 08:11:04 |
| 2017-07-31 | 2017-07-01 | | 36.89 | 191 | 4.99 | 19.27 | 84.08 | 08:00:41 |

**Figuur 32: Webscraping-dashboard - scherm *Overview***

Het dashboard bestaat uit 4 tabs: overview, global graphs, specific, specific graphs. Het overviewscherm geeft weer welke scripts gerund hebben en toont een samenvatting van de gescrapete gegevens: datum, maand, website, duurtijd van scrapen (in sec.), aantal verzamelde items, minimum prijs, gemiddelde prijs, maximum prijs, eindtijd van scrapen. Het tweede scherm 'Global Graphs' geeft een grafische voorstelling van elk script. Hieronder staat de gemiddelde prijsevolutie voor een boeking van een hotelverblijf (weekend):




**Figuur 33: Webscraping-dashboard - scherm *Global Graphs***

Het derde scherm ‘Specific’ geeft meer gedetailleerde informatie van een bepaald script, bijvoorbeeld voor de verschillende subcategorieën van een kledingwinkel:

Webscraping Dashboard | Overview | Global Graphs | Specific | Specific Graphs

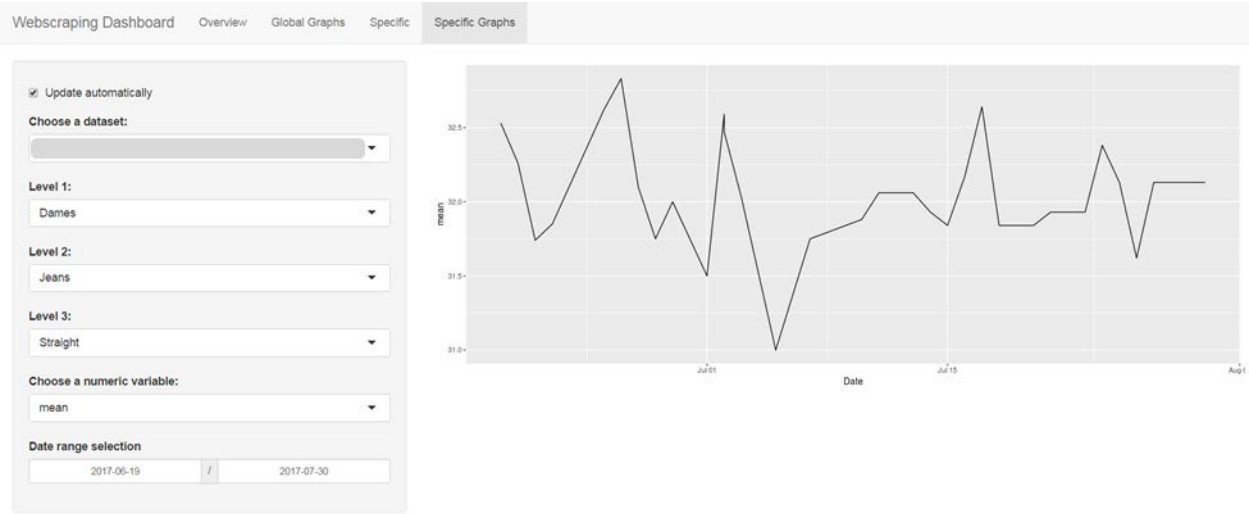
Site: | Month: All

Show 25 entries | Search:

| date | month | site | desc_cat_1 | desc_cat_2 | desc_cat_3 | count | min | mean | max |
|---|---|---|---|---|---|---|---|---|---|
| All | All | All | dames | jeans | All | All | All | All | All |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Zwangerschapsjeans | 19 | 29 | 34.79 | 39 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Jeans shorts | 13 | 9 | 20.54 | 39 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Jeggings | 16 | 9 | 11.69 | 19 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Bootcut & Flare | 7 | 29 | 36.14 | 39 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Straight | 30 | 19 | 32.13 | 39 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Slim | 14 | 19 | 31.57 | 39 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Super skinny | 10 | 19 | 22 | 29 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Skinny | 14 | 29 | 30.43 | 39 |
| 2017-07-30 | 2017-07-01 | | Dames | Jeans | Grote maten | 24 | 19 | 30.91 | 49 |
| 2017-07-29 | 2017-07-01 | | Dames | Jeans | Zwangerschapsjeans | 21 | 29 | 34.24 | 39 |
| 2017-07-29 | 2017-07-01 | | Dames | Jeans | Jeans shorts | 13 | 9 | 20.54 | 39 |
| 2017-07-29 | 2017-07-01 | | Dames | Jeans | Jeggings | 16 | 9 | 11.69 | 19 |

**Figuur 34: Webscraping-dashboard - scherm *Specific***

Het vierde scherm ‘Specific Graphs’ toont een visuele voorstelling van de specifieke resultaten:




**Figuur 35: Webscraping-dashboard - scherm *Specific Graphs***

Dagelijks wordt er ook automatisch een mail verstuurd met daarin de ‘*failed scripts*’. Aan de hand van de logs die worden aangemaakt tijdens het scrapen, kan er dan worden gecontroleerd wat er verkeerd is gelopen en indien nodig kan het script dan worden aangepast. Twee mogelijke *failures* zijn:

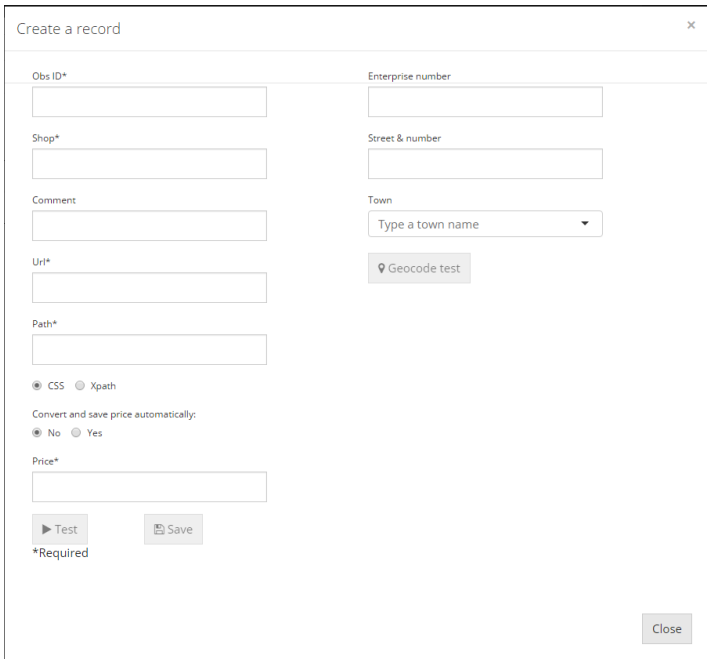
- Verbinding met server onderbroken, waardoor er geen connectie kon gemaakt worden met de betrokken website
- Website werd gewijzigd (html-code), waardoor het script niet meer volledig werkt

# 12. ROBOTTOOL

Voor producten die weinig van prijs veranderen of wanneer er slechts een aantal prijzen ingezameld moeten worden, is het niet aan te raden om scripts te schrijven en uit te voeren die de gegevens frequent ophalen. Dat zou niet enkel tijdrovend zijn wat betreft scriptontwikkeling (elke website is anders), maar ook de server zou hierdoor extra belast worden. Daarom werd een robottool ontwikkeld die automatisch kan nagaan of er op websites prijswijzigingen zijn. De tool kijkt voor een vooraf gespecifieerd item naar de website waar het product voorgesteld wordt en is geprogrammeerd om te controleren of de kenmerken van het item gewijzigd zijn ten opzichte van de vorige controle. Wanneer de robottool een verandering opmerkt, wordt dat aangeduid en kan er manueel nog nagegaan worden of de aangeduide wijziging correct is.

Technisch gezien volgt de robottool het '*navigation path*' zoals het product manueel zou worden opgezocht. De tool kijkt enkel naar de opgegeven kenmerken, voornamelijk de prijscontext. Anders zou de tool veel verschillen kunnen opmerken op de volledige webpagina, maar waar geen interesse in is. Door de focus te leggen op specifieke onderdelen (prijscontext) van de webpagina, verhoogt de efficiëntie van de tool. Uiteindelijk vergelijkt de robottool de prijscontext met het resultaat van de laatste uitvoering. Wanneer er een verschillende prijscontext is, worden de verschillen getoond aan de gebruikers van de robottool (markering). De verandering van prijscontext impliceert echter niet noodzakelijk dat de prijs ook effectief gewijzigd is. De gebruiker van de robottool kan dit dan controleren.

De volgende figuur toont het formulier waarmee een nieuwe check kan aangemaakt worden:




**Figuur 36: Formulier om nieuwe controle aan te maken**

Bij het creëren kan er ook gekozen worden om de prijswijziging, indien aanwezig, automatisch op te slaan als nieuwe prijs ('Convert and save price automatically').

Ter illustratie wordt een nieuwe check voor een bepaald product B gecreëerd. Bij het observeren van dit product op de website wordt een prijs van 9.50 euro genoteerd.

RobotTool | Overview | Product groups | Map

Test1 | Test | Data | Create | Refresh | Run

Search:

| Date | Obs ID | Shop | Comment | Price | Url |
|---|---|---|---|---|---|
| 04/08 | 2 | Telenet | Basic internet | 27.8 | https://www2.telenet.be/nl/internet-en-tv/internet/bestel-basic-internet/ |
| 04/08 | 45 | Test Product B | | 9.5 | https://kinepolis.be/nl/bioscopen/kinepolis-antwerpen/info |
| 04/08 | 5 | Belgacom | | 24.5 | https://www.proximus.be/nl/id_cr_int/particulieren/producten/internetabonnementen.html |

Showing 1 to 3 of 3 entries | Previous | 1 | Next

**Figuur 37: Voorbeeld Robottool - Obs ID 45 Test Product B**

Enige tijd later, bij het runnen van de tool, wordt volgend resultaat gemeld:

Results for observation Id: 45

| Shop | Comment | Last context | Old price | Url | Rgx | New context | New price |
|---|---|---|---|---|---|---|---|
| Test Product B | | € 9,50 | 9.5 | https://kinepolis.be/nl/bioscopen/kinepolis-antwerpen/info | 0 | € 10,90 | 9.5 |

RobotTool | Overview | Product groups | Map

Test1 | Test | Data | Refresh | Run

Search:

| Obs ID | Shop | Comment | Last context | Old price | Url | Rgx | New context | New price |
|---|---|---|---|---|---|---|---|---|
| 2 | Telenet | Basic internet | € 27,80 | 27.8 | https://www2.telenet.be/nl/internet-en-tv/internet/bestel-basic-internet/ | 1 | € 27,80 | 27.8 |
| 45 | Test Product B | | € 9,50 | 9.5 | https://kinepolis.be/nl/bioscopen/kinepolis-antwerpen/info | 0 | € 10,90 | 9.5 |
| 5 | Belgacom | | Vanaf € 27,50 /maand (incl. btw) | 24.5 | https://www.proximus.be/nl/id_cr_int/particulieren/producten/internetabonnementen.html | 0 | Vanaf € 27,50 /maand (incl. btw) | 24.5 |

Showing 1 to 3 of 3 entries | Previous | 1 | Next

**Figuur 38: Automatische markering van prijswijziging**

De tool meldt dus dat de prijscontext gewijzigd is. Na controle van het product op de website, wordt er inderdaad vastgesteld dat de prijs veranderd is en kan de nieuwe prijs in de robottool bevestigd worden.

Indien bij de creatie ‘Convert and save price automatically’ werd geactiveerd, is het resultaat:

RobotTool | Overview | Product groups | Map

Test1 | Test | Data | Refresh | Run

Search:

| Obs ID | Shop | Comment | Last context | Old price | Url | Rgx | New context | New price |
|---|---|---|---|---|---|---|---|---|
| 2 | Telenet | Basic internet | € 27,80 | 27.8 | https://www2.telenet.be/nl/internet-en-tv/internet/bestel-basic-internet/ | 1 | € 27,80 | 27.8 |
| 45 | Test Product B | | € 9,50 | 9.5 | https://kinepolis.be/nl/bioscopen/kinepolis-antwerpen/info | 1 | € 10,90 | 10.9 |
| 5 | Belgacom | | Vanaf € 27,50 /maand (incl. btw) | 24.5 | https://www.proximus.be/nl/id_cr_int/particulieren/producten/internetabonnementen.html | 0 | Vanaf € 27,50 /maand (incl. btw) | 24.5 |

Showing 1 to 3 of 3 entries | Previous | 1 | Next

**Figuur 39: Automatische markering van prijswijziging met aanduiding nieuwe prijs**

Waarbij de nieuwe context en prijs onmiddellijk getoond worden.

Hieronder staat een voorbeeld waarbij de prijscontext van ‘Test Product B’ veranderd is, maar niet de prijs. De oude context bevatte als beschrijving ‘Standaard tarief’, de nieuwe prijscontext vermeldt echter ‘Normaal tarief’. Dat wordt ook gedetecteerd door de robottool en wordt weergegeven voor de gebruikers. Die kunnen dan eventueel een extra controle doen.

Results for observation Id: 45

| Shop | Comment | Last context | Old price | Url | Rgx | New context | New price |
|---|---|---|---|---|---|---|---|
| Test Product B | | € 10,90 \| Standaard tarief | 10.9 | https://kinepolis.be/nl/bioscopen/kinepolis-antwerpen/info | 1 | € 10,90 \| Normaal tarief | 10.9 |

RobotTool

Overview | Product groups | Map

Test1 | Test | Refresh | Run

Search:

| Obs ID | Shop | Comment | Last context | Old price | Url | Rgx | New context | New price |
|---|---|---|---|---|---|---|---|---|
| 2 | Telenet | Basic internet | € 27,80 | 27.8 | https://www2.telenet.be/nl/internet-en-tv/internet/bestel-basic-internet/ | 1 | € 27,80 | 27.8 |
| 45 | Test Product B | | € 10,90 \| Standaard tarief | 10.9 | https://kinepolis.be/nl/bioscopen/kinepolis-antwerpen/info | 1 | € 10,90 \| Normaal tarief | 10.9 |
| 5 | Belgacom | | Vanaf € 27,50 /maand (incl. btw) | 24.5 | https://www.proximus.be/nl/id_cr_int/particulieren/producten/internetabonnementen.html | 0 | Vanaf € 27,50 /maand (incl. btw) | 24.5 |

Showing 1 to 3 of 3 entries

Previous | 1 | Next

**Figuur 40: Automatische markering van contextwijziging**

# 13. CONCLUSIE

In eerste instantie werd nagegaan voor welke items webscraping kon worden getest en toegepast. De belangrijkste beslissingsfactoren waren:

- items waarvoor reeds online prijzen werden opgezocht (bv. vliegtickets);
- de mogelijkheid om manuele prijscollectie te vervangen door webscraping als proxy voor de offline prijzen (bv. kleding);
- items die vaak online gekocht worden (bv. boeken, dvd's).

Momenteel bestaan er scripts voor vliegtickets, boeken, dvd's, videogames, 1 supermarkt, kleding, schoenen, elektronica (laptops, televisies, wasmachines,…), studentenkamers, hotels, internationale treintickets, tweedehandswagens,…

De resultaten van de supermarkt, Blu-ray, dvd en internationale treinreizen worden reeds gebruikt binnen de berekening van CPI en HICP. Voor de andere segmenten worden op regelmatige tijdstippen gegevens gescrapet en verzameld om verder onderzoek uit te voeren. Voor een aantal van deze segmenten die zich in de onderzoeksfase bevinden werden de resultaten en de berekeningswijzen van indexcijfers in dit verslag aan de hand van case studies weergegeven.

De case studie voor schoenen toonde aan dat via webscraping prijsindexcijfers kunnen berekend worden op basis van online ingezamelde gegevens die sterk aanleunen tegen deze die berekend worden op basis van prijzen genoteerd in fysieke winkels. De sectie over de hotels gaf weer dat door webscraping een steekproef makkelijk kan worden uitgebreid (meer dan 1.000 keer groter). De studies van de studentenkamers en de tweedehandswagens maakten duidelijk dat via webscraping mogelijks nieuwe consumptiesegmenten gecapteerd kunnen worden die voordien moeilijk op te nemen waren in de indexkorf. Uit het onderdeel over consumentenelektronica bleek dat hedonische methoden misschien mogelijk worden voor een aantal producten omdat nu ook productkenmerken geregistreerd kunnen worden in combinatie met een uitgebreide steekproef.

Ook werd duidelijk dat machine learning soms nodig is. Enerzijds om bepaalde ongewenste observaties te filteren zoals de studie van dvd's en Blu-rays aantoonde, anderzijds omdat via webscraping er te veel observaties zijn waardoor het onmogelijk is om ze manueel te klasseren. Machine learning kan ook hier helpen zoals bleek uit de case over kleding.

Uiteraard is het de bedoeling om het aantal websites dat gescrapet wordt verder uit te breiden. En is continue opvolging van de scripts nodig (bv. bij wijziging van de structuur van een website). Daarvoor werd het dashboard ontwikkeld. Voor sites waar slechts een aantal prijzen moet worden gescrapet werd een robottool ontwikkeld.

# OVER STATBEL

Statbel, het Belgische statistiekbureau, verzamelt, produceert en verspreidt betrouwbare en pertinente cijfers over de Belgische economie, de samenleving en het territorium.

Het verzamelen gebeurt op basis van administratieve databronnen en enquêtes, het produceren gebeurt op een wetenschappelijke, kwaliteitsvolle wijze en de statistieken worden tijdig en op een klantvriendelijke wijze verspreid.

De privacy, de bescherming van vertrouwelijke gegevens en het gebruik voor louter statistische doeleinden worden hierbij gegarandeerd.

**Bezoek onze website**
www.statbel.fgov.be

**Of contacteer ons**
e-mail: statbel@economie.fgov.be

Statbel (Algemene Directie Statistiek - Statistics Belgium)
North Gate - Koning Albert II-laan, 16 - 1000 Brussel
E-mail: statbel@economie.fgov.be

Ondernemingsnummer
0314.595.348

Verantwoordelijke uitgever
Nico Waeyaert

North Gate
Koning Albert II-laan 16
1000 Brussel