Supplementary material

Overture data product attributes

Table i: Attributes in Overture UK open data product, with an additional indicator describing which attributes were processed out of their nested JSON format.

Attribute	Description	Processed
id	Unique ID assigned by Overture.	N
updatetime, version	Information about the version (and date) of the Overture data - most recent.	N
confidence	Attribute assigned by Overture	N
websites, socials, emails, phones	Contact information and social media associated with POI.	N
names_value	POI name.	Y
category_main, category_alternative	Assigned category for POI.	Y
addresses_postcode, addresses_freeform, addresses_country, addresses_locality, addresses_region	Address information for POI.	Y
sources_dataset	Data source (meta or microsoft)	N
lat, lng	Coordinates for POI.	N
h3_01 ... h3_09	H3 addresses for each POI.	N
easting, northing	Coordiantes for POI.	N
LAD22CD ... LGD2014_nm	Administrative geographies for each POI, different for England/Wales, Scotland and Northern Ireland	N

Scraping Overture data for specific brands

As discussed in the main body, extracting a complete list of POIs for a brand (e.g., Waitrose) is difficult due to missing values in the category and brand columns. For example, selecting all POIs by brand, as below in Figure i, results in 0 POIs for Waitrose, and 1,934 POIs for Spar.

```r
library(arrow)
library(sf)
library(tidyverse)

## Open dataset with arrow
places <- open_dataset("data/uk_places_admin.parquet")

## Filter to Waitrose stores - by brand only [RETURNS 0 POIS]
waitrose <- places %>%
  filter(brand_name_value == "Waitrose") %>%
  as.data.frame()

## Filter to Spar stores - by brand only [RETURNS 1,934 POIS]
spar <- places %>%
  filter(brand_name_value == "SPAR" | brand_name_value == "Spar") %>%
  as.data.frame()
```

Figure i: R code snippet highlighting brand filtering of POIs.

However, by integrating additional filters, specifically on the POI name and the POI category columns, you can get to the final list of POIs for each retail brand, which closely resemble those in the Geolytix dataset. For example, in Figure ii we show that the brand and name columns need to be filtered to extract all Spar supermarket stores, and name and category columns need to be filtered to extract all Waitrose supermarkets (excluding petrol stations and other retail formats). This code snippet (Figure ii) results in 420 POIs for Waitrose, and 2,308 POIs for Spar, as in Table 1. This has strong implications for future use of this dataset, where users should consider POI name, brand and category if seeking to extract a complete list of POIs for specific research questions or applications.

```r
library(arrow)
library(sf)
library(tidyverse)

## Open dataset with arrow
places <- open_dataset("data/uk_places_admin.parquet")

## Filter to Waitrose stores - by brand only [RETURNS 420 POIS]
waitrose <- places %>%
  filter(str_detect(names_value, "Waitrose")) %>%
  filter(category_main == "supermarket" | category_main == "grocery_store" | category_main == "convenience_store") %>%
  as.data.frame()

## Filter to Spar stores - by brand only [RETURNS 2,308 POIS]
spar <- places %>%
  filter(names_value == "Spar" | names_value == "SPAR" | brand_name_value == "SPAR") %>%
  as.data.frame()
```

Figure ii: Code snippet highlighting additional filtering of POI attributes.