

## Purpose

The purpose of this paper is to layout in mathematical terms an approach to calculating population forecasts for various types of geographical regions within the United Kingdom

## Background

There are 3 types of geographical regions of concern.

LAD - Local Authority Districts.

OA - Output areas, the smallest area covered by an official UK census.

CA - Catchment areas which are in principle an arbitrary geographical area but usually represented by a geographical location (latitude, longitude) and a radius.

The data we have to work with is as follows:

## Projected populations by LAD, age and year

At some point in the past projections, or estimates, were made of what the population was expected to be for each LAD in the United Kingdom, for each age between 0 and 100, and for each year between 2018 and 2035 inclusive.

Those projections can best be thought of (for the purpose of this paper) as a function `pp(lad, age, year)` which returns a non-negative integer as a result.

So for example if `pp('E09000002', 24, 2027)` gave the answer `123,456` this would be telling us that in 2027 (Jan 1st to be precise) the projected population of 24 year olds in the Barking and Dagenham Local Authorities District is `123456`.

## Actual population data for LAD

We also have, apparently, actual population data, by age, for all LADs as at 2023. Presumably this data comes from a recent census.

Putting this in mathematical terms, this data provides us with a function which I will call `pc(lad, age)` which returns a non-negative integer.

So for example if `pc('E09000002', 24)` gave the answer `345,543` this would tell us that according to the census the actual population of 24 year olds, in Barking & Dagenham was at the time of the census in 2023 `345,543`. Notice that the `pc()` function does not take a year parameter.

## How to calculate forward looking population forecasts for LADs

The `pp()` function allows us to calculate population growth by age for each year between 2018 and 2025. The equation for this is:

$$g(lad, age, year) = pp(lad, age, year) / pp(lad, age, 2023)$$

Further by combining the `pp()` function and the `pc()` function we can make new estimates of population by age (as a function called `ep()`) factoring in the growth rates embedded in `pp()` and the latest census data embedded in `pc()`. The equation for this is as follows:

$$ep(lad, age, year) = g(lad, age, year) * pc(lad, age)$$

which is equivalent to

$$ep(lad, age, year) = (pp(lad, age, year)/pp(lad, age, 2023)) * pc(lad, age)$$

## What about catchment areas ?

Fortunately we have some additional data available that will assist us calculate estimated population for catchment areas. And as above I will represent this additional data as function.

The first is a simple function that I will call `cn(lat, lng, radius)` which returns the name of the **most important** (probably the biggest) output area OA, contained within the catchment area.

The second piece of information tells us what LADs overlap a catchment area and the fraction of the LAD population inside the catchment area.

This is also represented by functions, but its a little more complicated.

This time there are 2 function the first I call `contribution(lat, lng, radius)` for it gives me a list of LADs that contribute to a catchment area, and the second function is called `share(lat, lng, radius, lad)`. What it returns is the share of the population of the LAD with code `lad` inside the catchment area. The share is always a number between 0.0 and 1.0. The last parameter of `share()` is only permitted to be a LAD code that was in the list returned by the contribution function.

### First step for CAs

Synthesize values that are equivalent to the `pp()` function for LAD data.

Lets assume the `lat, lng, radius` are fixed for the duration of this discussion.

$$pp(lat, lng, radius, age, year) = \sum_{lad \in contribution(lat, lng, radius)} pp(lad, age, year) \times share(lat, lng, radius, lad)$$

and an analagous equation for `pc(lat, lng, radius, age)`

$$pc(lat, lng, radius, age) = \sum_{lad \in contribution(lat, lng, radius)} pc(lad, age) \times share(lat, lng, radius, lad)$$

Hence we can now calculate `ep()` for the catchment area as

```
ep(lat, lng, age, year) = (pp(lat, lng, age, year) / pp(lat, lng, age,
2023)) * pc(lat, lng, radius, age)
```

To put this more into programming terms:

```
pp(lat, lng, radius, age, year) => {
  v = 0
  for lad in contribution(lat, lng, radius) {
    v = v + share(lat, lng, radius, age) * pp(lad, age, year)
  }
  return v
}

pc(lat, lng, radius, age) => {
  v = 0
  for lad in contribution(lat, lng, radius) {
    v = v + share(lat, lng, radius, age) * pc(lad, age)
  }
}

ep(lat, lng, radius, age, year) => {
  (pp(lat, lng, radius, age, year) / pp(lat, lng, age, 2023)) * pc(lat,
lng, age)
}
```