Mapping Nutritional Inequality: A Spatial Analysis of Food Purchases Across MSOA Regions in Greater London

Aksa Johnson

2025-07-03

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)
library(readr)
library(scales)
```

```
##
## Attaching package: 'scales'

## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
library(cols4all)
```

```
## Warning: package 'cols4all' was built under R version 4.4.3
```

```r
data <- read_csv("Sep_msoa_grocery.csv")
```

```
## Rows: 977 Columns: 202
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (1): area_id
## dbl (201): weight, weight_perc2.5, weight_perc25, weight_perc50, weight_perc...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
data
```

```
## # A tibble: 977 x 202
##    area_id   weight weight_perc2.5 weight_perc25 weight_perc50 weight_perc75
##    <chr>      <dbl>          <dbl>         <dbl>         <dbl>         <dbl>
##  1 E02000001   315.           37             150           250           400
##  2 E02000002   415.           32.5           160           300           500
##  3 E02000003   421.           32.5           154           320           500
##  4 E02000004   349.           30             125           265           450
##  5 E02000005   429.           30             157           300           500
##  6 E02000007   388.           32.5           150           300           500
##  7 E02000008   399.           32.5           151           300           500
##  8 E02000009   399.           30             150           300           500
##  9 E02000010   409.           32.5           150           300           494
## 10 E02000011   385.           25             150           300           470
## # i 967 more rows
## # i 196 more variables: weight_perc97.5 <dbl>, weight_std <dbl>,
## #   weight_ci95 <dbl>, volume <dbl>, volume_perc2.5 <dbl>, volume_perc25 <dbl>,
## #   volume_perc50 <dbl>, volume_perc75 <dbl>, volume_perc97.5 <dbl>,
## #   volume_std <dbl>, volume_ci95 <dbl>, fat <dbl>, fat_perc2.5 <dbl>,
## #   fat_perc25 <dbl>, fat_perc50 <dbl>, fat_perc75 <dbl>, fat_perc97.5 <dbl>,
## #   fat_std <dbl>, fat_ci95 <dbl>, saturate <dbl>, saturate_perc2.5 <dbl>, ...
```

```r
#view column names and structure
str(data)
```

```
## spc_tbl_ [977 x 202] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ area_id                 : chr [1:977] "E02000001" "E02000002" "E02000003" "E02000004" ...
##  $ weight                  : num [1:977] 315 415 421 349 429 ...
##  $ weight_perc2.5          : num [1:977] 37 32.5 32.5 30 30 32.5 32.5 30 32.5 25 ...
##  $ weight_perc25           : num [1:977] 150 160 154 125 157 150 151 150 150 150 ...
##  $ weight_perc50           : num [1:977] 250 300 320 265 300 300 300 300 300 300 ...
##  $ weight_perc75           : num [1:977] 400 500 500 450 500 500 500 500 494 470 ...
##  $ weight_perc97.5         : num [1:977] 1000 1380 1500 1000 1500 ...
##  $ weight_std              : num [1:977] 305 610 643 415 679 ...
##  $ weight_ci95             : num [1:977] 3.23 21.68 17.55 15.21 20.88 ...
##  $ volume                  : num [1:977] 100 111 121 102 125 ...
##  $ volume_perc2.5          : num [1:977] 10.5 18.2 15 16 11 ...
##  $ volume_perc25           : num [1:977] 50 45.5 35.5 50 40 ...
##  $ volume_perc50           : num [1:977] 75 80 75 75 75 75 75 100 75 70 ...
##  $ volume_perc75           : num [1:977] 114 150 160 114 200 ...
##  $ volume_perc97.5         : num [1:977] 264 341 400 341 341 ...
##  $ volume_std              : num [1:977] 87.1 100.5 139.9 88.1 116.9 ...
##  $ volume_ci95             : num [1:977] 1.94 8.38 8.68 7.38 8.15 ...
##  $ fat                     : num [1:977] 8.71 9.12 8.75 8.78 8.81 ...
##  $ fat_perc2.5             : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ fat_perc25              : num [1:977] 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 ...
##  $ fat_perc50              : num [1:977] 1.8 2.5 2.1 2.8 2 2.6 2.6 2.7 2.2 2.6 ...
##  $ fat_perc75              : num [1:977] 13.2 14.1 13.5 14.5 13.6 ...
##  $ fat_perc97.5            : num [1:977] 46.8 42.6 45 34.9 44 ...
##  $ fat_std                 : num [1:977] 13.9 13.9 13.5 12 13.7 ...
##  $ fat_ci95                : num [1:977] 0.119 0.394 0.298 0.349 0.337 ...
##  $ saturate                : num [1:977] 3.53 3.55 3.32 3.34 3.49 ...
```

```
##  $ saturate_perc2.5        : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ saturate_perc25         : num [1:977] 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 ...
##  $ saturate_perc50         : num [1:977] 0.5 0.8 0.7 1 0.6 0.9 0.9 0.9 0.8 0.9 ...
##  $ saturate_perc75         : num [1:977] 3.7 4.1 3.5 4.1 4.1 4.1 4.5 4.1 4.1 4.5 ...
##  $ saturate_perc97.5       : num [1:977] 21.3 19.6 19 18.5 19.5 ...
##  $ saturate_std            : num [1:977] 6.48 6.38 6.08 5.3 6.19 ...
##  $ saturate_ci95           : num [1:977] 0.0554 0.1801 0.1348 0.1538 0.1516 ...
##  $ salt                    : num [1:977] 0.55 0.522 0.497 0.6 0.534 ...
##  $ salt_perc2.5            : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ salt_perc25             : num [1:977] 0.01 0.05 0.05 0.05 0.01 0.05 0.01 0.03 0.02 0.01 ...
##  $ salt_perc50             : num [1:977] 0.11 0.22 0.18 0.26 0.13 0.3 0.2 0.24 0.2 0.2 ...
##  $ salt_perc75             : num [1:977] 0.7 0.8 0.782 0.8 0.7 ...
##  $ salt_perc97.5           : num [1:977] 2.6 2.2 2 2.13 2.13 ...
##  $ salt_std                : num [1:977] 4.65 1.25 1.05 1.39 1.81 ...
##  $ salt_ci95               : num [1:977] 0.0398 0.0352 0.0233 0.0405 0.0444 ...
##  $ sugar                   : num [1:977] 9.7 11.2 11.4 11 11.9 ...
##  $ sugar_perc2.5           : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ sugar_perc25            : num [1:977] 1 1.4 1.5 1.3 1.4 1.2 1.4 1.1 1.3 1.2 ...
##  $ sugar_perc50            : num [1:977] 3.5 4.4 4 3.7 4.6 4.1 4.2 3.7 3.8 4 ...
##  $ sugar_perc75            : num [1:977] 10 11.8 11.8 11.8 12.7 ...
##  $ sugar_perc97.5          : num [1:977] 57 58.3 59.9 58.5 59.9 ...
##  $ sugar_std               : num [1:977] 15.7 16.8 17.4 17 17.7 ...
##  $ sugar_ci95              : num [1:977] 0.134 0.473 0.386 0.495 0.435 ...
##  $ protein                 : num [1:977] 5.36 5.11 5.34 5.71 4.76 ...
##  $ protein_perc2.5         : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ protein_perc25          : num [1:977] 0.6 0.6 0.5 0.6 0.4 0.4 0.4 0.4 0.4 0.3 ...
##  $ protein_perc50          : num [1:977] 2.8 3.2 3.5 3.6 2.4 3.3 3 3.2 2.9 2.2 ...
##  $ protein_perc75          : num [1:977] 7.8 7.9 8.5 8.8 7.4 ...
##  $ protein_perc97.5        : num [1:977] 24 22.5 21.9 23.3 21.9 ...
##  $ protein_std             : num [1:977] 6.73 5.94 5.95 6.32 5.88 ...
##  $ protein_ci95            : num [1:977] 0.0576 0.1677 0.1318 0.1835 0.144 ...
##  $ carb                    : num [1:977] 16 21.2 23.7 20.7 19.5 ...
##  $ carb_perc2.5            : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ carb_perc25             : num [1:977] 0.5 1.5 1.8 1.5 0.3 0.4 0.2 0.2 1.2 0.1 ...
##  $ carb_perc50             : num [1:977] 5.6 10.9 11.8 11.3 8.9 10.9 9.6 9 10 7 ...
##  $ carb_perc75             : num [1:977] 20.9 38.4 45.6 30.9 30.3 ...
##  $ carb_perc97.5           : num [1:977] 71.3 75.1 80.4 73.5 73.5 ...
##  $ carb_std                : num [1:977] 21.6 24.1 26.5 23.7 23.7 ...
##  $ carb_ci95               : num [1:977] 0.184 0.679 0.586 0.687 0.582 ...
##  $ fibre                   : num [1:977] 1.64 1.68 1.74 1.69 1.55 ...
##  $ fibre_perc2.5           : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ fibre_perc25            : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ fibre_perc50            : num [1:977] 1 1.1 1 1.1 1 1 1 1 1 0.9 ...
##  $ fibre_perc75            : num [1:977] 2.2 2.3 2.4 2.3 2.2 2.2 2.2 2.2 2.2 2.2 ...
##  $ fibre_perc97.5          : num [1:977] 8 7.06 8.5 8 6.9 ...
##  $ fibre_std               : num [1:977] 2.56 2.67 2.59 2.36 2.33 ...
##  $ fibre_ci95              : num [1:977] 0.0219 0.0753 0.0575 0.0685 0.057 ...
##  $ alcohol                 : num [1:977] 0.299 0.131 0.226 0.126 0.129 ...
##  $ alcohol_perc2.5         : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ alcohol_perc25          : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ alcohol_perc50          : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ alcohol_perc75          : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ alcohol_perc97.5        : num [1:977] 7.04 0 0 0 0 0 0 0 0 0 ...
##  $ alcohol_std             : num [1:977] 1.76 1.43 2.13 1.23 1.37 ...
```

```
##  $ alcohol_ci95            : num [1:977] 0.0151 0.0403 0.0472 0.0357 0.0336 ...
##  $ energy_fat              : num [1:977] 78.4 82.1 78.8 79 79.3 ...
##  $ energy_fat_perc2.5      : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ energy_fat_perc25       : num [1:977] 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 1.8 ...
##  $ energy_fat_perc50       : num [1:977] 16.2 22.5 18.9 25.2 18 23.4 23.4 24.3 19.8 23.4 ...
##  $ energy_fat_perc75       : num [1:977] 119 127 122 130 123 ...
##  $ energy_fat_perc97.5     : num [1:977] 421 383 405 314 396 ...
##  $ energy_fat_std          : num [1:977] 125 126 121 108 124 ...
##  $ energy_fat_ci95         : num [1:977] 1.07 3.54 2.69 3.14 3.03 ...
##  $ energy_saturate         : num [1:977] 31.8 31.9 29.9 30 31.4 ...
##  $ energy_saturate_perc2.5 : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##  $ energy_saturate_perc25  : num [1:977] 0.9 0.9 0.9 0.9 0.9 0.9 0.9 0.9 0.9 0.9 ...
##  $ energy_saturate_perc50  : num [1:977] 4.5 7.2 6.3 9 5.4 8.1 8.1 8.1 7.2 8.1 ...
##  $ energy_saturate_perc75  : num [1:977] 33.3 36.9 31.5 36.9 36.9 36.9 40.5 36.9 36.9 40.5 ...
##  $ energy_saturate_perc97.5 : num [1:977] 192 176 171 166 175 ...
##  $ energy_saturate_std     : num [1:977] 58.3 57.5 54.8 47.7 55.7 ...
##  $ energy_saturate_ci95    : num [1:977] 0.499 1.621 1.213 1.385 1.364 ...
##  $ energy_sugar            : num [1:977] 38.8 44.8 45.5 44.1 47.7 ...
##  $ energy_sugar_perc2.5    : num [1:977] 0 0 0 0 0 0 0 0 0 0 ...
##   [list output truncated]
##  - attr(*, "spec")=
##   .. cols(
##   ..   area_id = col_character(),
##   ..   weight = col_double(),
##   ..   weight_perc2.5 = col_double(),
##   ..   weight_perc25 = col_double(),
##   ..   weight_perc50 = col_double(),
##   ..   weight_perc75 = col_double(),
##   ..   weight_perc97.5 = col_double(),
##   ..   weight_std = col_double(),
##   ..   weight_ci95 = col_double(),
##   ..   volume = col_double(),
##   ..   volume_perc2.5 = col_double(),
##   ..   volume_perc25 = col_double(),
##   ..   volume_perc50 = col_double(),
##   ..   volume_perc75 = col_double(),
##   ..   volume_perc97.5 = col_double(),
##   ..   volume_std = col_double(),
##   ..   volume_ci95 = col_double(),
##   ..   fat = col_double(),
##   ..   fat_perc2.5 = col_double(),
##   ..   fat_perc25 = col_double(),
##   ..   fat_perc50 = col_double(),
##   ..   fat_perc75 = col_double(),
##   ..   fat_perc97.5 = col_double(),
##   ..   fat_std = col_double(),
##   ..   fat_ci95 = col_double(),
##   ..   saturate = col_double(),
##   ..   saturate_perc2.5 = col_double(),
##   ..   saturate_perc25 = col_double(),
##   ..   saturate_perc50 = col_double(),
##   ..   saturate_perc75 = col_double(),
##   ..   saturate_perc97.5 = col_double(),
##   ..   saturate_std = col_double(),
```

```
##   ..   saturate_ci95 = col_double(),
##   ..   salt = col_double(),
##   ..   salt_perc2.5 = col_double(),
##   ..   salt_perc25 = col_double(),
##   ..   salt_perc50 = col_double(),
##   ..   salt_perc75 = col_double(),
##   ..   salt_perc97.5 = col_double(),
##   ..   salt_std = col_double(),
##   ..   salt_ci95 = col_double(),
##   ..   sugar = col_double(),
##   ..   sugar_perc2.5 = col_double(),
##   ..   sugar_perc25 = col_double(),
##   ..   sugar_perc50 = col_double(),
##   ..   sugar_perc75 = col_double(),
##   ..   sugar_perc97.5 = col_double(),
##   ..   sugar_std = col_double(),
##   ..   sugar_ci95 = col_double(),
##   ..   protein = col_double(),
##   ..   protein_perc2.5 = col_double(),
##   ..   protein_perc25 = col_double(),
##   ..   protein_perc50 = col_double(),
##   ..   protein_perc75 = col_double(),
##   ..   protein_perc97.5 = col_double(),
##   ..   protein_std = col_double(),
##   ..   protein_ci95 = col_double(),
##   ..   carb = col_double(),
##   ..   carb_perc2.5 = col_double(),
##   ..   carb_perc25 = col_double(),
##   ..   carb_perc50 = col_double(),
##   ..   carb_perc75 = col_double(),
##   ..   carb_perc97.5 = col_double(),
##   ..   carb_std = col_double(),
##   ..   carb_ci95 = col_double(),
##   ..   fibre = col_double(),
##   ..   fibre_perc2.5 = col_double(),
##   ..   fibre_perc25 = col_double(),
##   ..   fibre_perc50 = col_double(),
##   ..   fibre_perc75 = col_double(),
##   ..   fibre_perc97.5 = col_double(),
##   ..   fibre_std = col_double(),
##   ..   fibre_ci95 = col_double(),
##   ..   alcohol = col_double(),
##   ..   alcohol_perc2.5 = col_double(),
##   ..   alcohol_perc25 = col_double(),
##   ..   alcohol_perc50 = col_double(),
##   ..   alcohol_perc75 = col_double(),
##   ..   alcohol_perc97.5 = col_double(),
##   ..   alcohol_std = col_double(),
##   ..   alcohol_ci95 = col_double(),
##   ..   energy_fat = col_double(),
##   ..   energy_fat_perc2.5 = col_double(),
##   ..   energy_fat_perc25 = col_double(),
##   ..   energy_fat_perc50 = col_double(),
##   ..   energy_fat_perc75 = col_double(),
```

```
##   ..   energy_fat_perc97.5 = col_double(),
##   ..   energy_fat_std = col_double(),
##   ..   energy_fat_ci95 = col_double(),
##   ..   energy_saturate = col_double(),
##   ..   energy_saturate_perc2.5 = col_double(),
##   ..   energy_saturate_perc25 = col_double(),
##   ..   energy_saturate_perc50 = col_double(),
##   ..   energy_saturate_perc75 = col_double(),
##   ..   energy_saturate_perc97.5 = col_double(),
##   ..   energy_saturate_std = col_double(),
##   ..   energy_saturate_ci95 = col_double(),
##   ..   energy_sugar = col_double(),
##   ..   energy_sugar_perc2.5 = col_double(),
##   ..   energy_sugar_perc25 = col_double(),
##   ..   energy_sugar_perc50 = col_double(),
##   ..   energy_sugar_perc75 = col_double(),
##   ..   energy_sugar_perc97.5 = col_double(),
##   ..   energy_sugar_std = col_double(),
##   ..   energy_sugar_ci95 = col_double(),
##   ..   energy_protein = col_double(),
##   ..   energy_protein_perc2.5 = col_double(),
##   ..   energy_protein_perc25 = col_double(),
##   ..   energy_protein_perc50 = col_double(),
##   ..   energy_protein_perc75 = col_double(),
##   ..   energy_protein_perc97.5 = col_double(),
##   ..   energy_protein_std = col_double(),
##   ..   energy_protein_ci95 = col_double(),
##   ..   energy_carb = col_double(),
##   ..   energy_carb_perc2.5 = col_double(),
##   ..   energy_carb_perc25 = col_double(),
##   ..   energy_carb_perc50 = col_double(),
##   ..   energy_carb_perc75 = col_double(),
##   ..   energy_carb_perc97.5 = col_double(),
##   ..   energy_carb_std = col_double(),
##   ..   energy_carb_ci95 = col_double(),
##   ..   energy_fibre = col_double(),
##   ..   energy_fibre_perc2.5 = col_double(),
##   ..   energy_fibre_perc25 = col_double(),
##   ..   energy_fibre_perc50 = col_double(),
##   ..   energy_fibre_perc75 = col_double(),
##   ..   energy_fibre_perc97.5 = col_double(),
##   ..   energy_fibre_std = col_double(),
##   ..   energy_fibre_ci95 = col_double(),
##   ..   energy_alcohol = col_double(),
##   ..   energy_alcohol_perc2.5 = col_double(),
##   ..   energy_alcohol_perc25 = col_double(),
##   ..   energy_alcohol_perc50 = col_double(),
##   ..   energy_alcohol_perc75 = col_double(),
##   ..   energy_alcohol_perc97.5 = col_double(),
##   ..   energy_alcohol_std = col_double(),
##   ..   energy_alcohol_ci95 = col_double(),
##   ..   energy_tot = col_double(),
##   ..   energy_tot_perc2.5 = col_double(),
##   ..   energy_tot_perc25 = col_double(),
```

```
##   ..   energy_tot_perc50 = col_double(),
##   ..   energy_tot_perc75 = col_double(),
##   ..   energy_tot_perc97.5 = col_double(),
##   ..   energy_tot_std = col_double(),
##   ..   energy_tot_ci95 = col_double(),
##   ..   f_energy_fat = col_double(),
##   ..   f_energy_saturate = col_double(),
##   ..   f_energy_sugar = col_double(),
##   ..   f_energy_protein = col_double(),
##   ..   f_energy_carb = col_double(),
##   ..   f_energy_fibre = col_double(),
##   ..   f_energy_alcohol = col_double(),
##   ..   energy_density = col_double(),
##   ..   h_nutrients_weight = col_double(),
##   ..   h_nutrients_weight_norm = col_double(),
##   ..   h_nutrients_calories = col_double(),
##   ..   h_nutrients_calories_norm = col_double(),
##   ..   f_beer = col_double(),
##   ..   f_dairy = col_double(),
##   ..   f_eggs = col_double(),
##   ..   f_fats_oils = col_double(),
##   ..   f_fish = col_double(),
##   ..   f_fruit_veg = col_double(),
##   ..   f_grains = col_double(),
##   ..   f_meat_red = col_double(),
##   ..   f_poultry = col_double(),
##   ..   f_readymade = col_double(),
##   ..   f_sauces = col_double(),
##   ..   f_soft_drinks = col_double(),
##   ..   f_spirits = col_double(),
##   ..   f_sweets = col_double(),
##   ..   f_tea_coffee = col_double(),
##   ..   f_water = col_double(),
##   ..   f_wine = col_double(),
##   ..   f_dairy_weight = col_double(),
##   ..   f_eggs_weight = col_double(),
##   ..   f_fats_oils_weight = col_double(),
##   ..   f_fish_weight = col_double(),
##   ..   f_fruit_veg_weight = col_double(),
##   ..   f_grains_weight = col_double(),
##   ..   f_meat_red_weight = col_double(),
##   ..   f_poultry_weight = col_double(),
##   ..   f_readymade_weight = col_double(),
##   ..   f_sauces_weight = col_double(),
##   ..   f_sweets_weight = col_double(),
##   ..   h_items = col_double(),
##   ..   h_items_norm = col_double(),
##   ..   h_items_weight = col_double(),
##   ..   h_items_weight_norm = col_double(),
##   ..   representativeness_norm = col_double(),
##   ..   transaction_days = col_double(),
##   ..   num_transactions = col_double(),
##   ..   man_day = col_double(),
##   ..   population = col_double(),
```

```
##   ..   male = col_double(),
##   ..   female = col_double(),
##   ..   age_0_17 = col_double(),
##   ..   age_18_64 = col_double(),
##   ..   `age_65+` = col_double(),
##   ..   avg_age = col_double(),
##   ..   area_sq_km = col_double(),
##   ..   people_per_sq_km = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
head(data)
```

```
## # A tibble: 6 x 202
##   area_id   weight weight_perc2.5 weight_perc25 weight_perc50 weight_perc75
##   <chr>      <dbl>          <dbl>         <dbl>         <dbl>         <dbl>
## 1 E02000001   315.           37             150           250           400
## 2 E02000002   415.           32.5           160           300           500
## 3 E02000003   421.           32.5           154           320           500
## 4 E02000004   349.           30             125           265           450
## 5 E02000005   429.           30             157           300           500
## 6 E02000007   388.           32.5           150           300           500
## # i 196 more variables: weight_perc97.5 <dbl>, weight_std <dbl>,
## #   weight_ci95 <dbl>, volume <dbl>, volume_perc2.5 <dbl>, volume_perc25 <dbl>,
## #   volume_perc50 <dbl>, volume_perc75 <dbl>, volume_perc97.5 <dbl>,
## #   volume_std <dbl>, volume_ci95 <dbl>, fat <dbl>, fat_perc2.5 <dbl>,
## #   fat_perc25 <dbl>, fat_perc50 <dbl>, fat_perc75 <dbl>, fat_perc97.5 <dbl>,
## #   fat_std <dbl>, fat_ci95 <dbl>, saturate <dbl>, saturate_perc2.5 <dbl>,
## #   saturate_perc25 <dbl>, saturate_perc50 <dbl>, saturate_perc75 <dbl>, ...
```

```
names(data)
```

```
##   [1] "area_id"                     "weight"
##   [3] "weight_perc2.5"              "weight_perc25"
##   [5] "weight_perc50"               "weight_perc75"
##   [7] "weight_perc97.5"             "weight_std"
##   [9] "weight_ci95"                 "volume"
##  [11] "volume_perc2.5"              "volume_perc25"
##  [13] "volume_perc50"               "volume_perc75"
##  [15] "volume_perc97.5"             "volume_std"
##  [17] "volume_ci95"                 "fat"
##  [19] "fat_perc2.5"                 "fat_perc25"
##  [21] "fat_perc50"                  "fat_perc75"
##  [23] "fat_perc97.5"                "fat_std"
##  [25] "fat_ci95"                    "saturate"
##  [27] "saturate_perc2.5"            "saturate_perc25"
##  [29] "saturate_perc50"             "saturate_perc75"
##  [31] "saturate_perc97.5"           "saturate_std"
##  [33] "saturate_ci95"               "salt"
##  [35] "salt_perc2.5"                "salt_perc25"
##  [37] "salt_perc50"                 "salt_perc75"
##  [39] "salt_perc97.5"               "salt_std"
##  [41] "salt_ci95"                   "sugar"
```

```
##  [43] "sugar_perc2.5"             "sugar_perc25"
##  [45] "sugar_perc50"              "sugar_perc75"
##  [47] "sugar_perc97.5"            "sugar_std"
##  [49] "sugar_ci95"                "protein"
##  [51] "protein_perc2.5"           "protein_perc25"
##  [53] "protein_perc50"            "protein_perc75"
##  [55] "protein_perc97.5"          "protein_std"
##  [57] "protein_ci95"              "carb"
##  [59] "carb_perc2.5"              "carb_perc25"
##  [61] "carb_perc50"               "carb_perc75"
##  [63] "carb_perc97.5"             "carb_std"
##  [65] "carb_ci95"                 "fibre"
##  [67] "fibre_perc2.5"             "fibre_perc25"
##  [69] "fibre_perc50"              "fibre_perc75"
##  [71] "fibre_perc97.5"            "fibre_std"
##  [73] "fibre_ci95"                "alcohol"
##  [75] "alcohol_perc2.5"           "alcohol_perc25"
##  [77] "alcohol_perc50"            "alcohol_perc75"
##  [79] "alcohol_perc97.5"          "alcohol_std"
##  [81] "alcohol_ci95"              "energy_fat"
##  [83] "energy_fat_perc2.5"        "energy_fat_perc25"
##  [85] "energy_fat_perc50"         "energy_fat_perc75"
##  [87] "energy_fat_perc97.5"       "energy_fat_std"
##  [89] "energy_fat_ci95"           "energy_saturate"
##  [91] "energy_saturate_perc2.5"   "energy_saturate_perc25"
##  [93] "energy_saturate_perc50"    "energy_saturate_perc75"
##  [95] "energy_saturate_perc97.5"  "energy_saturate_std"
##  [97] "energy_saturate_ci95"      "energy_sugar"
##  [99] "energy_sugar_perc2.5"      "energy_sugar_perc25"
## [101] "energy_sugar_perc50"       "energy_sugar_perc75"
## [103] "energy_sugar_perc97.5"     "energy_sugar_std"
## [105] "energy_sugar_ci95"         "energy_protein"
## [107] "energy_protein_perc2.5"    "energy_protein_perc25"
## [109] "energy_protein_perc50"     "energy_protein_perc75"
## [111] "energy_protein_perc97.5"   "energy_protein_std"
## [113] "energy_protein_ci95"       "energy_carb"
## [115] "energy_carb_perc2.5"       "energy_carb_perc25"
## [117] "energy_carb_perc50"        "energy_carb_perc75"
## [119] "energy_carb_perc97.5"      "energy_carb_std"
## [121] "energy_carb_ci95"          "energy_fibre"
## [123] "energy_fibre_perc2.5"      "energy_fibre_perc25"
## [125] "energy_fibre_perc50"       "energy_fibre_perc75"
## [127] "energy_fibre_perc97.5"     "energy_fibre_std"
## [129] "energy_fibre_ci95"         "energy_alcohol"
## [131] "energy_alcohol_perc2.5"    "energy_alcohol_perc25"
## [133] "energy_alcohol_perc50"     "energy_alcohol_perc75"
## [135] "energy_alcohol_perc97.5"   "energy_alcohol_std"
## [137] "energy_alcohol_ci95"       "energy_tot"
## [139] "energy_tot_perc2.5"        "energy_tot_perc25"
## [141] "energy_tot_perc50"         "energy_tot_perc75"
## [143] "energy_tot_perc97.5"       "energy_tot_std"
## [145] "energy_tot_ci95"           "f_energy_fat"
## [147] "f_energy_saturate"         "f_energy_sugar"
## [149] "f_energy_protein"          "f_energy_carb"
```

```
## [151] "f_energy_fibre"            "f_energy_alcohol"
## [153] "energy_density"            "h_nutrients_weight"
## [155] "h_nutrients_weight_norm"   "h_nutrients_calories"
## [157] "h_nutrients_calories_norm" "f_beer"
## [159] "f_dairy"                   "f_eggs"
## [161] "f_fats_oils"               "f_fish"
## [163] "f_fruit_veg"               "f_grains"
## [165] "f_meat_red"                "f_poultry"
## [167] "f_readymade"               "f_sauces"
## [169] "f_soft_drinks"             "f_spirits"
## [171] "f_sweets"                  "f_tea_coffee"
## [173] "f_water"                   "f_wine"
## [175] "f_dairy_weight"            "f_eggs_weight"
## [177] "f_fats_oils_weight"        "f_fish_weight"
## [179] "f_fruit_veg_weight"        "f_grains_weight"
## [181] "f_meat_red_weight"         "f_poultry_weight"
## [183] "f_readymade_weight"        "f_sauces_weight"
## [185] "f_sweets_weight"           "h_items"
## [187] "h_items_norm"              "h_items_weight"
## [189] "h_items_weight_norm"       "representativeness_norm"
## [191] "transaction_days"          "num_transactions"
## [193] "man_day"                   "population"
## [195] "male"                      "female"
## [197] "age_0_17"                  "age_18_64"
## [199] "age_65+"                   "avg_age"
## [201] "area_sq_km"                "people_per_sq_km"
```

```r
#checking for missing values
any(colSums(is.na(data)) > 0)
```

```
## [1] FALSE
```

To compare the nutritional quality of food purchases across MSOAs, I created a factor Nutritional quality Score(NQS) by combining multiple nutritional indicators into a single index. The variables included total energy, sugar, saturated fat, fibre and nutrient diversity. These variables are selected in evidence-based dietary guidelines provided by World Health Organisation(WHO) and some articles about dietary requirements. These organisations identify high intake of sugar, saturated fat, and energy(calories) as risk factor for obesity and non-communicable diseases, including diabetes, heart diseases, stroke and cancer.On the other hand, fibre and nutrient diversity are linked to overall diet quality and protective benefits. These variables were standardized using z scores to ensure that they are on same scale which wont be affect the final score.

```r
#standardize each variable

data <- data%>%
  mutate(
    z_calories = scale(energy_tot),
    z_sugar = scale(sugar),
    z_sat_fat = scale(saturate),
    z_fibre = scale(fibre),
    z_nutrient_diversity = scale(h_nutrients_weight_norm)
    )
```

Computing Nutritional Quality Score The NQS formula is adapted from similar composite methods used in public health. Nutrients linked to healthy diets, such as fibre and nutrient diversity, contributed positively to the quality score, whereas components that are often recommeneded for limited intake, such as, sugars, saturated fat, and total energy, contributed negatively. (The Nutri-Score: A Science-Based Front-of-Pack Nutrition Label 2021)

For computing Nutrient Quality Score, fibre and nutrient diversity were positively weighted in the score to reflect their contribution to healthy diets, while enrgy, sugar, fat were neagtively weighted due to their contribution with poor dietary outcomes. The resulting NQS provides a composite measure of diet quality for each, allowing for meaningful spatial comparison and visualisation.

```
 data <-  data%>%
   mutate(
     NQS = z_fibre +z_nutrient_diversity - z_sat_fat - z_sugar - z_calories
   )
head(data$NQS)
```

```
##             [,1]
## [1,]  3.240370
## [2,] -3.606438
## [3,] -3.753462
## [4,] -1.597517
## [5,] -3.988615
## [6,] -6.346609
```

Compute proportions of age groups

```
#
data <- data %>%
  mutate(
    prop_children = age_0_17 / population,
    prop_adults = age_18_64 / population,
    prop_elderly = `age_65+` / population
  )
```

Calculate age-specific NQS values

```
data <- data %>%
  mutate(
    NQS_children = NQS * prop_children,
    NQS_adults = NQS * prop_adults,
    NQS_elderly = NQS * prop_elderly
  )
head(data %>% select(area_id, NQS, NQS_children, NQS_adults, NQS_elderly))
```

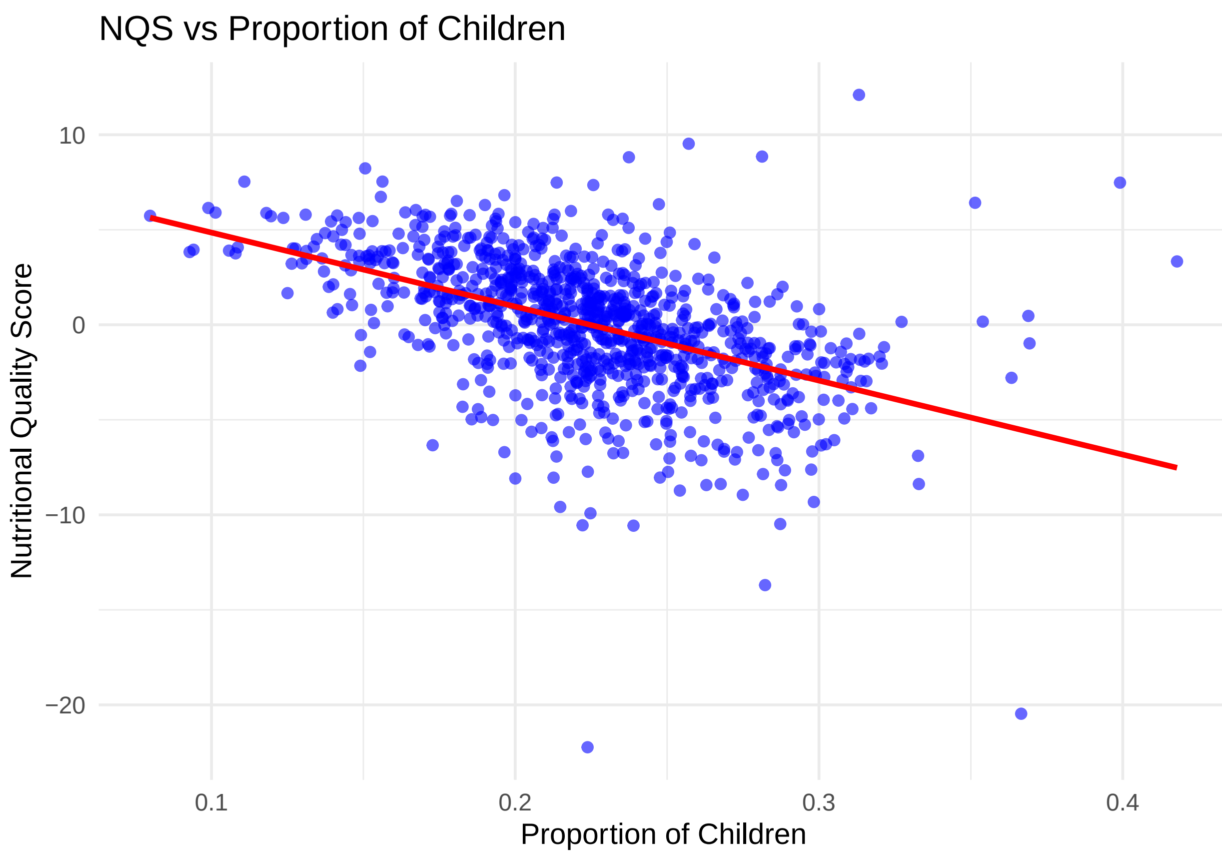
```
## # A tibble: 6 x 5
##   area_id   NQS[,1] NQS_children[,1] NQS_adults[,1] NQS_elderly[,1]
##   <chr>       <dbl>            <dbl>          <dbl>           <dbl>
## 1 E02000001    3.24            0.493          2.11            0.640
## 2 E02000002   -3.61           -1.05          -2.03           -0.527
## 3 E02000003   -3.75           -0.967         -2.33           -0.458
## 4 E02000004   -1.60           -0.355         -0.988          -0.254
## 5 E02000005   -3.99           -1.16          -2.40           -0.436
## 6 E02000007   -6.35           -1.91          -3.73           -0.707
```

```r
#Standardize it
data <- data%>%
  mutate(
    z_NQS_children = scale(NQS_children),
    z_NQS_adults = scale(NQS_adults),
    z_NQS_elderly = scale(NQS_elderly)
    )
head(data %>% select(area_id, NQS, z_NQS_children, z_NQS_adults, z_NQS_elderly))
```

```
## # A tibble: 6 x 5
##   area_id   NQS[,1] z_NQS_children[,1] z_NQS_adults[,1] z_NQS_elderly[,1]
##   <chr>       <dbl>              <dbl>            <dbl>             <dbl>
## 1 E02000001    3.24              0.691            0.900              1.46
## 2 E02000002   -3.61             -1.18            -0.940             -1.17
## 3 E02000003   -3.75             -1.08            -1.07              -1.01
## 4 E02000004   -1.60             -0.340           -0.477             -0.552
## 5 E02000005   -3.99             -1.31            -1.10              -0.963
## 6 E02000007   -6.35             -2.23            -1.70              -1.57
```

```r
# Children vs NQS
ggplot(data, aes(x = prop_children, y = NQS)) +
  geom_point(alpha = 0.6, color = "blue") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "NQS vs Proportion of Children", x = "Proportion of Children", y = "Nutritional Quality S
  theme_minimal()
```

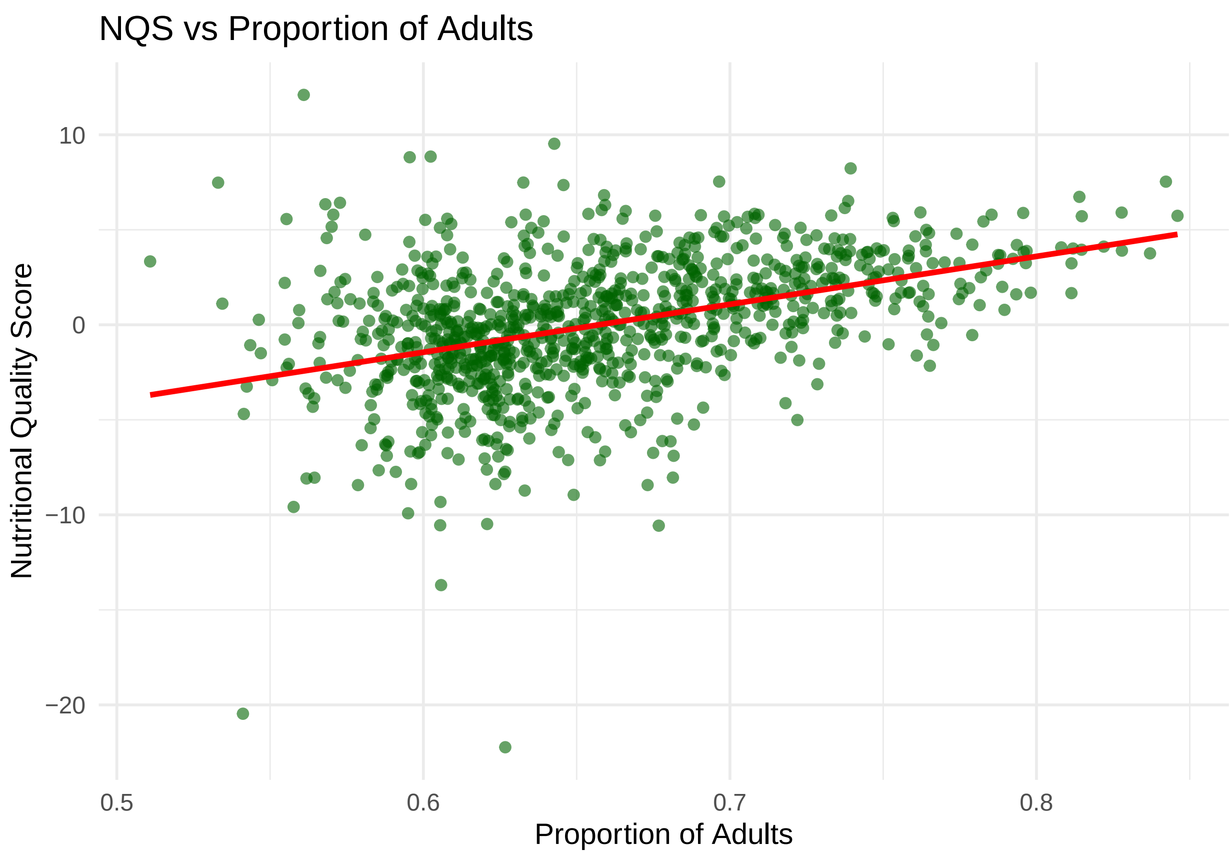
```
## `geom_smooth()` using formula = 'y ~ x'
```



This plot shows a downward trend.There is a negative relationship between the proportion of children and the NQS. That means, the more children in an area, the lower the nutritional quality of food.Therefore, it is essential to examine the main NQS and weight it by age group to assess the nutritional vulnerability of children across London.

```
# Adults vs NQS
ggplot(data, aes(x = prop_adults, y = NQS)) +
  geom_point(alpha = 0.6, color = "darkgreen") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "NQS vs Proportion of Adults", x = "Proportion of Adults", y = "Nutritional Quality Scor
  theme_minimal()
```

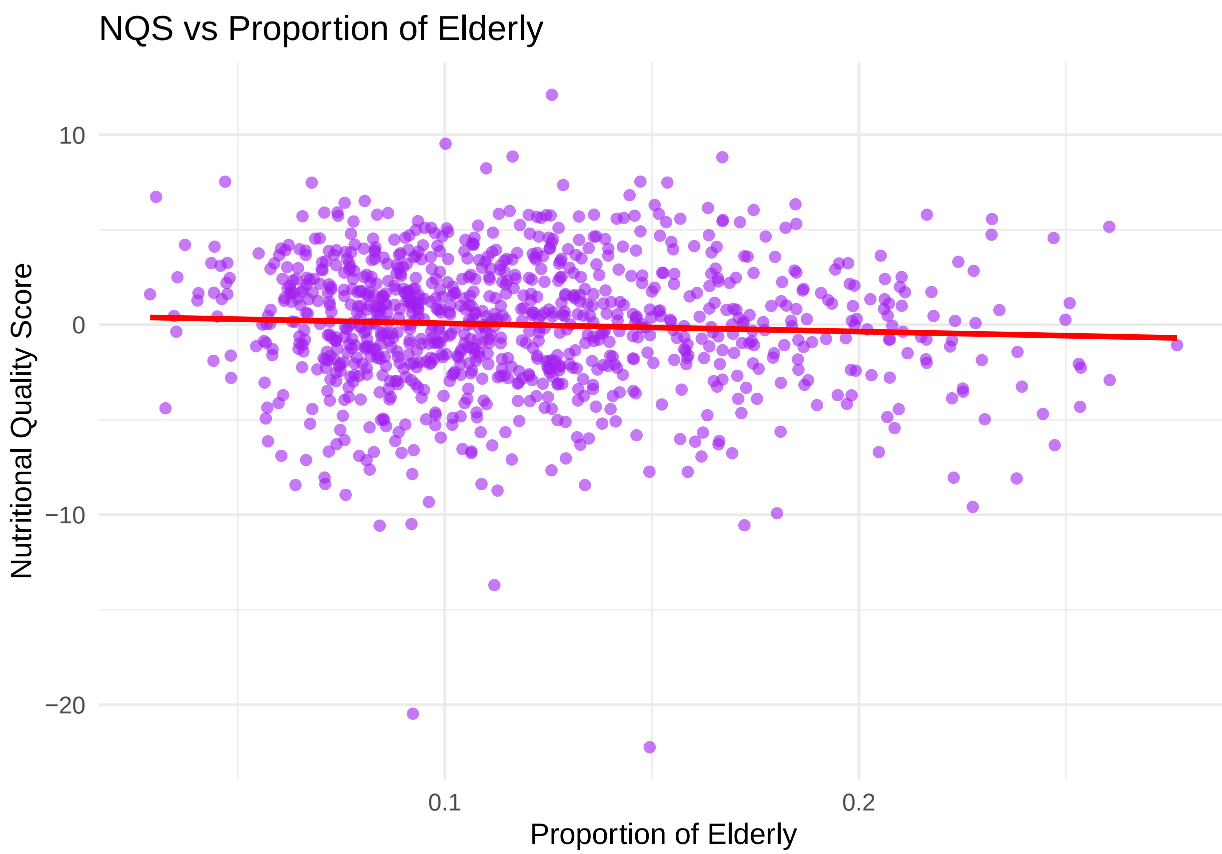
```
## `geom_smooth()` using formula = 'y ~ x'
```



This scatterplot shows a positive relatuonship between adults and NQS.That means the areas with more adults (18–64 years) tend to have better nutritional quality in their food purchases.This trend contrasts with the negative association observed in areas with more children, underlining potential age-related disparities in dietary choices or food access.

```
# Elderly vs NQS
ggplot(data, aes(x = prop_elderly, y = NQS)) +
  geom_point(alpha = 0.6, color = "purple") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "NQS vs Proportion of Elderly", x = "Proportion of Elderly", y = "Nutritional Quality Sc
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

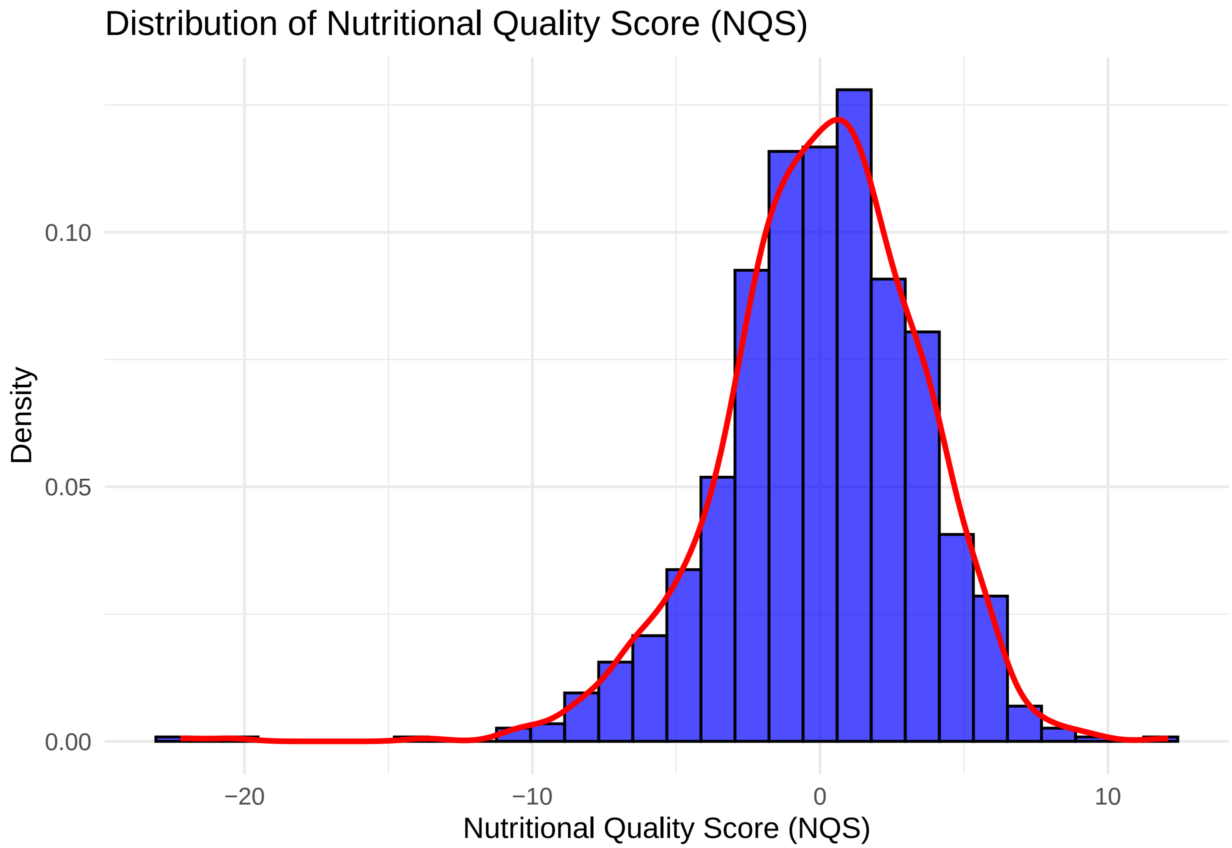


The red regression line is almost flat.This shows no clear relationship between the proportion of elderly and the Nutritional Quality Score (NQS).

```
ggplot(data, aes(x = NQS)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  geom_density(color = "red", size = 1) +
  labs(
    title = "Distribution of Nutritional Quality Score (NQS)",
    x = "Nutritional Quality Score (NQS)",
    y = "Density"
  ) +
  theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



The distribution of the Nutritional Quality Score (NQS) across MSOAs in Greater London follows an approximately normal pattern, with the majority of areas falling near the average. However, there are several regions with significantly low NQS values, highlighting pockets of poor nutritional purchasing behavior. These regions warrant further attention, especially when a high proportion of children reside in them. ###Creating Maps

```
pacman::p_load(tidyverse,dplyr,readr,sf,tmap,here)
packageVersion("tmap")
```

```
## [1] '4.1'
```

```
year_msoa_grocery <- data%>%
  rename(msoa_code = area_id)
msoas <-st_read(here("statistical-gis-boundaries-london/ESRI/MSOA_2011_London_gen_MHW.shp")) %>%rename(
```

```
## Reading layer 'MSOA_2011_London_gen_MHW' from data source
##   'C:\Users\aksaj\OneDrive\Documents\PROJECT\statistical-gis-boundaries-london\ESRI\MSOA_2011_London_
##   using driver 'ESRI Shapefile'
## Simple feature collection with 983 features and 12 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: 503574.2 ymin: 155850.8 xmax: 561956.7 ymax: 200933.6
## Projected CRS: OSGB36 / British National Grid
```

```r
tesco_and_msoas <- inner_join(msoas,year_msoa_grocery)
```

```
## Joining with ‘by = join_by(msoa_code)‘
```

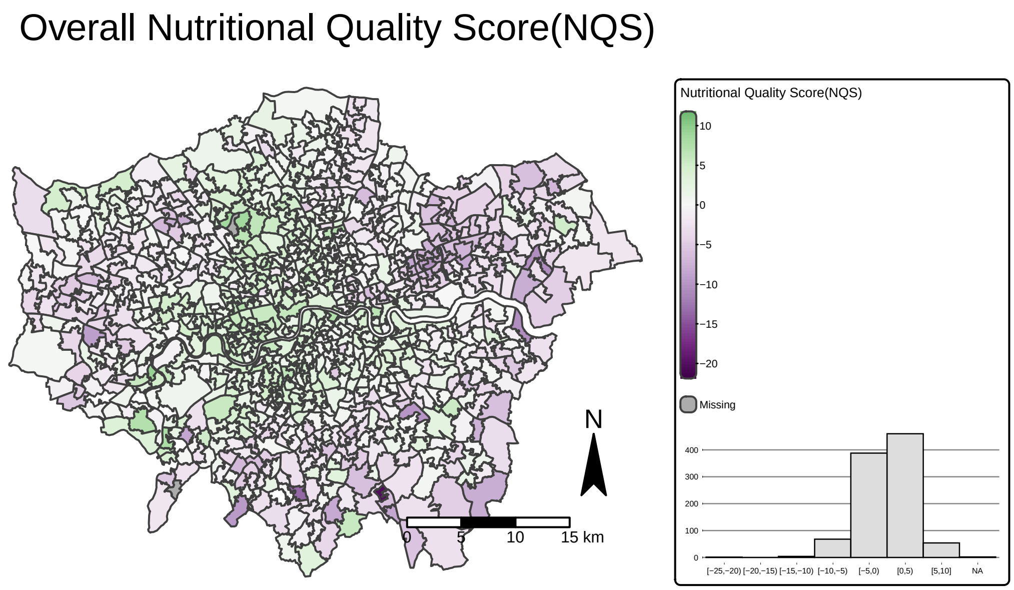
##Overall NQS Map

```r
tmap_mode("plot")  # use "view" for interactive mode
```

```
## i tmap mode set to "plot".
```

```r
tm_shape(tesco_and_msoas) +
  tm_polygons(
    fill = "NQS",
    fill.chart = tm_chart_histogram(),
    fill.legend = tm_legend(title = "Nutritional Quality Score(NQS)", reverse = TRUE),
    fill.scale = tm_scale_continuous(limits = c(-22,12),values = "brewer.PRGn")
  ) +
  tm_title(" Overall Nutritional Quality Score(NQS)") +
  tm_compass() +
  tm_scalebar() +
  tm_layout(frame = FALSE, bg.color = "white")
```

```
## Values have been found that are lower than the lowest limit. These ’outliers’ have been set to NA. Us
## Values have been found that are higher than the upper limit. These ’outliers’ have been set to NA. Us
## Variable(s) "fill" contains positive and negative values, so midpoint is set to 0. Set midpoint = NA
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## ‘cols4all::c4a_gui()‘ to explore them. The old palette name "brewer.PRGn" is
## named "prgn" (in long format "brewer.prgn")[plot mode] fit legend/component: Some legend items or map
## fit well, and are therefore rescaled.
## i Set the tmap option ‘component.autoscale = FALSE‘ to disable rescaling.
```



This map presents the overall diet quality of food purchases across Greater London.

High NQS (dark green): Predominantly in central, west, and southwest London, indicating healthier purchasing behaviour - higher fibre, greater nutrient diversity, and lower sugar, saturated fat, and calories.

Low NQS (purple): Concentrated in eastern and southeastern London, suggesting more processed and less diverse food purchases. White to pale shades: Areas close to the London average approximately NQS= 0.

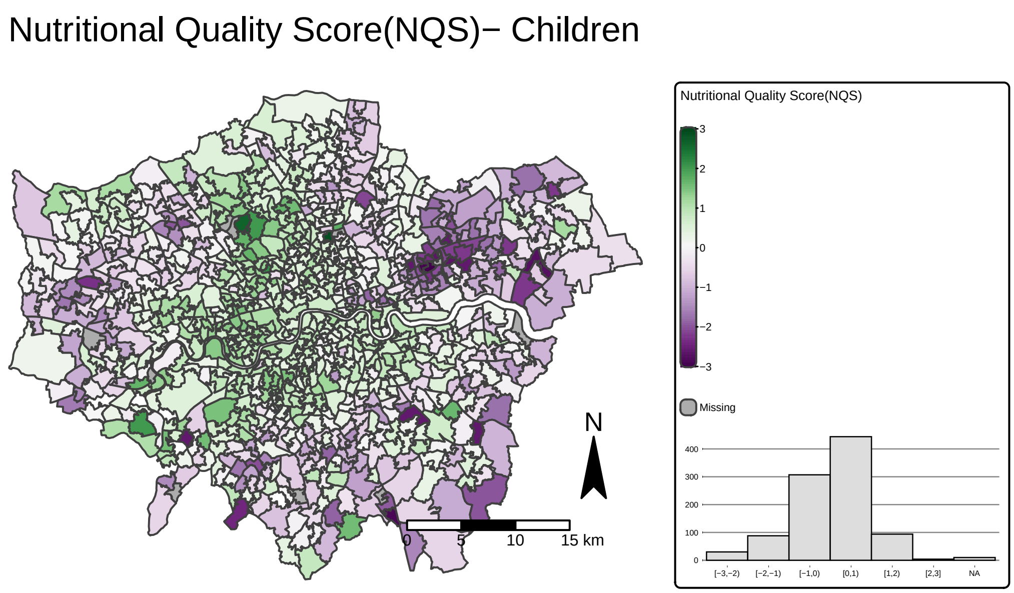
Grey (Missing): Areas without NQS data due to no matching purchase records or values beyond the set scale limits. The spatial pattern reveals a nutritional divide, with healthier purchasing trends more common in wealthier, better-connected areas.

```
tmap_mode("plot")  # use "view" for interactive mode
```

```
## i tmap mode set to "plot".
```

```
tm_shape(tesco_and_msoas) +
  tm_polygons(
    fill = "z_NQS_children",
    fill.chart = tm_chart_histogram(),
    fill.legend = tm_legend(title = "Nutritional Quality Score(NQS)", reverse = TRUE),
    fill.scale = tm_scale_continuous(limits = c(-3,3),values = "brewer.PRGn")
    ) +
  tm_title("Nutritional Quality Score(NQS)- Children") +
  tm_compass() +
  tm_scalebar() +
  tm_layout(frame = FALSE, bg.color = "white")
```

```
## Values have been found that are lower than the lowest limit. These ’outliers’ have been set to NA. Us
## Values have been found that are higher than the upper limit. These ’outliers’ have been set to NA. Us
## Variable(s) "fill" contains positive and negative values, so midpoint is set to 0. Set midpoint = NA
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## ‘cols4all::c4a_gui()‘ to explore them. The old palette name "brewer.PRGn" is
## named "prgn" (in long format "brewer.prgn")[plot mode] fit legend/component: Some legend items or map
## fit well, and are therefore rescaled.
## i Set the tmap option ‘component.autoscale = FALSE‘ to disable rescaling.
```



This map isolates the nutritional quality of purchases weighted by the proportion of children (aged 0-17) in each MSOA.

High NQS (dark green): Found mostly in west and southwest London, potentially reflecting higher parental health awareness, better access to fresh produce, and healthier retail environments for families.

Low NQS (Purple): Concentrated in east and southeast London, possibly linked to affordability constraints, higher exposure to fast food outlets, or limited access to varied, nutritious foods. White to pale shades mark areas near the London average, while grey - Missing zones highlight MSOAs with insufficient data for calculation.

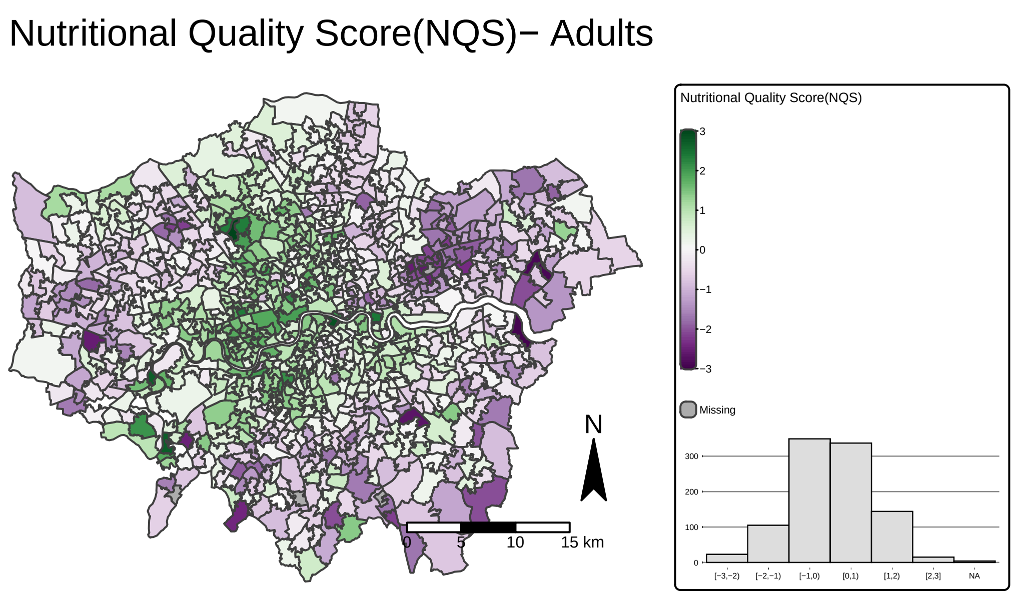
These patterns suggest children in some regions are disproportionately exposed to lower-quality diets, a public health concern given the long-term impacts on growth and wellbeing.

```r
tmap_mode("plot")  # use "view" for interactive mode
```

```
## i tmap mode set to "plot".
```

```
tm_shape(tesco_and_msoas) +
  tm_polygons(
    fill = "z_NQS_adults",
    fill.chart = tm_chart_histogram(),
    fill.legend = tm_legend(title = "Nutritional Quality Score(NQS)", reverse = TRUE),
    fill.scale = tm_scale_continuous(limits = c(-3,3),values = "brewer.PRGn")
  ) +
  tm_title("Nutritional Quality Score(NQS)- Adults") +
  tm_compass() +
  tm_scalebar() +
  tm_layout(frame = FALSE, bg.color = "white")
```

```
## Values have been found that are lower than the lowest limit. These 'outliers' have been set to NA. U
## Variable(s) "fill" contains positive and negative values, so midpoint is set to 0. Set midpoint = NA
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## 'cols4all::c4a_gui()' to explore them. The old palette name "brewer.PRGn" is
## named "prgn" (in long format "brewer.prgn")[plot mode] fit legend/component: Some legend items or ma
## fit well, and are therefore rescaled.
## i Set the tmap option 'component.autoscale = FALSE' to disable rescaling.
```



This map focuses on purchases weighted by the proportion of adults (18-64 years).

High NQS (dark green): Especially visible in central and northwestern MSOAs, where working-age adults appear to make healthier dietary choices. This could be influenced by higher disposable incomes, workplace wellness culture, or better supermarket provision.

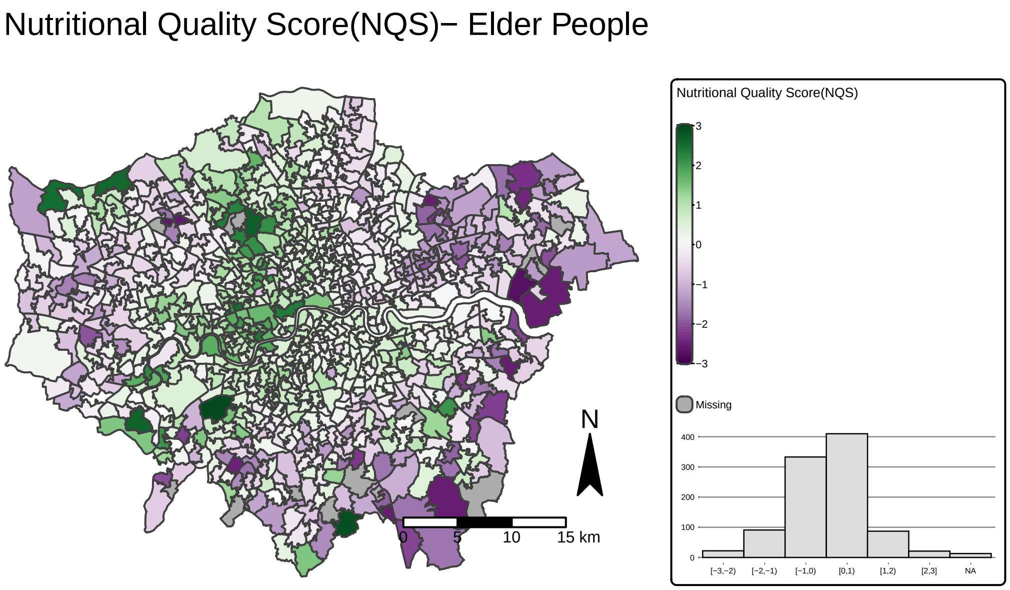
Low NQS (Purple): Scattered pockets across outer boroughs, where adults may face affordability barriers or opt for convenience foods due to lifestyle demands. White to pale shades mark areas near the London average, while grey - Missing zones highlight MSOAs with insufficient data for calculation. Compared to children's NQS, adult scores tend to be higher overall, suggesting adults generally purchase healthier food than other age groups.

```r
tmap_mode("plot")  # use "view" for interactive mode
```

```
## i tmap mode set to "plot".
```

```r
tm_shape(tesco_and_msoas) +
  tm_polygons(
    fill = "z_NQS_elderly",
    fill.chart = tm_chart_histogram(),
    fill.legend = tm_legend(title = "Nutritional Quality Score(NQS)", reverse = TRUE),
    fill.scale = tm_scale_continuous(limits = c(-3,3),values = "brewer.PRGn")
  ) +
  tm_title("Nutritional Quality Score(NQS)- Elder People") +
  tm_compass() +
  tm_scalebar() +
  tm_layout(frame = FALSE, bg.color = "white")
```

```
## Values have been found that are lower than the lowest limit. These 'outliers' have been set to NA. U
## Values have been found that are higher than the upper limit. These 'outliers' have been set to NA. U
## Variable(s) "fill" contains positive and negative values, so midpoint is set to 0. Set midpoint = NA
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## `cols4all::c4a_gui()` to explore them. The old palette name "brewer.PRGn" is
## named "prgn" (in long format "brewer.prgn")[plot mode] fit legend/component: Some legend items or ma
## fit well, and are therefore rescaled.
## i Set the tmap option `component.autoscale = FALSE` to disable rescaling.
```



This map weights the NQS by the proportion of elderly residents (65+ years).

High NQS (dark green): Found in certain central and northern areas, suggesting older residents there maintain healthier purchasing habits, possibly due to established dietary routines and access to quality food outlets.

Low NQS (Purple): More common in peripheral and some southern MSOAs, where older residents may face physical access challenges, fixed incomes, or reliance on less healthy nearby shops.

White to pale shades mark areas near the London average, while grey - Missing zones highlight MSOAs with insufficient data for calculation.

We can observe that several central and northern MSOAs demonstrate relatively better nutritional outcomes for the elderly, while some peripheral or southern areas show lower NQS values. This variation may reflect differences in access to nutritious food, socioeconomic status, or health awareness among older residents. Identifying such patterns is essential to target dietary interventions where older populations may be at higher nutritional risk.

After visualising the spatial distribution of the Nutritional Quality Score (NQS) using a choropleth map, the next step is to statistically assess whether the observed spatial patterns are random or exhibit clustering.

To do this, I will perform spatial autocorrelation analysis using Moran's I, which measures whether MSOAs with similar NQS values are geographically clustered. A positive and statistically significant Moran's I value would indicate clustering of similar NQS scores.