Short Paper

Alice Anonymous[*,a], Bob Security[b], Cat Memes[**,b], Derek Zoolander[**,a]

[a] *Department, Street, City, State, Zip*
[b] *Department, Street, City, State, Zip*

Abstract

This is the abstract.

It consists of two paragraphs.

1. Introduction

```
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
```

```
library(readr)
library(tidyr)
library(ggplot2)
library(ggrepel)

# this is a lookup table matching MSOAs to major towns and cities
city_names <- read_csv(’../data-raw/Middle_Layer_Super_Output_Area__2011__to_Major_Towns_and
```

```
## Parsed with column specification:
## cols(
##   MSOA11CD = col_character(),
```

[*] Corresponding Author

[**] Equal contribution

Email addresses: alice@example.com (Alice Anonymous), bob@example.com (Bob Security), cat@example.com (Cat Memes), derek@example.com (Derek Zoolander)

Preprint submitted to An awesome journal *October 17, 2020*

```
##   MSOA11NM = col_character(),
##   TCITY15CD = col_character(),
##   TCITY15NM = col_character(),
##   FID = col_double()
## )
```

```r
# change column name
city_names <- city_names %>% rename(city = TCITY15NM)

#unique cities
unique(city_names$city)
```

```
##   [1] "London"               NA                     "Bolton"
##   [4] "Stockport"            "Rochdale"             "Salford"
##   [7] "Manchester"           "Oldham"               "Bury"
##  [10] "Wigan"                "Rotherham"            "Liverpool"
##  [13] "Doncaster"            "Sheffield"            "St Helens"
##  [16] "Barnsley"             "Southport"            "Birkenhead"
##  [19] "Birmingham"           "Coventry"             "Dudley"
##  [22] "Newcastle upon Tyne"  "Sunderland"           "Sutton Coldfield"
##  [25] "South Shields"        "West Bromwich"        "Gateshead"
##  [28] "Leeds"                "Wolverhampton"        "Bradford"
##  [31] "Walsall"              "Huddersfield"         "Wakefield"
##  [34] "Solihull"             "Halifax"              "Leicester"
##  [37] "Nottingham"           "Hartlepool"           "Derby"
##  [40] "Blackpool"            "Grimsby"              "Warrington"
##  [43] "Scunthorpe"           "Kingston upon Hull"   "York"
##  [46] "Blackburn"            "Middlesbrough"        "Stockton-on-Tees"
##  [49] "Darlington"           "Telford"              "Stoke-on-Trent"
##  [52] "Swindon"              "Bristol"              "Peterborough"
##  [55] "Luton"                "Southend-on-Sea"      "Gillingham"
##  [58] "Plymouth"             "Weston-Super-Mare"    "Bath"
##  [61] "Bournemouth"          "Poole"                "Portsmouth"
##  [64] "Milton Keynes"        "Southampton"          "Bedford"
##  [67] "Reading"              "Chatham"              "Bracknell"
##  [70] "Slough"               "High Wycombe"         "Brighton and Hove"
##  [73] "Cambridge"            "Carlisle"             "Chester"
##  [76] "Chesterfield"         "Chelmsford"           "Basildon"
##  [79] "Eastbourne"           "Hastings"             "Exeter"
##  [82] "Colchester"           "Harlow"               "Cheltenham"
##  [85] "Gloucester"           "Hemel Hempstead"      "St Albans"
##  [88] "Stevenage"            "Basingstoke"          "Burnley"
##  [91] "Watford"              "Maidstone"            "Preston"
##  [94] "Norwich"              "Lincoln"              "Harrogate"
##  [97] "Northampton"          "Newcastle-under-Lyme" "Oxford"
## [100] "Mansfield"            "Shrewsbury"           "Burton upon Trent"
## [103] "Woking"                 "Guildford"              "Ipswich"
## [106] "Crawley"                "Nuneaton"               "Worthing"
## [109] "Redditch"               "Worcester"              "Swansea"
## [112] "Cardiff"                "Newport"
```

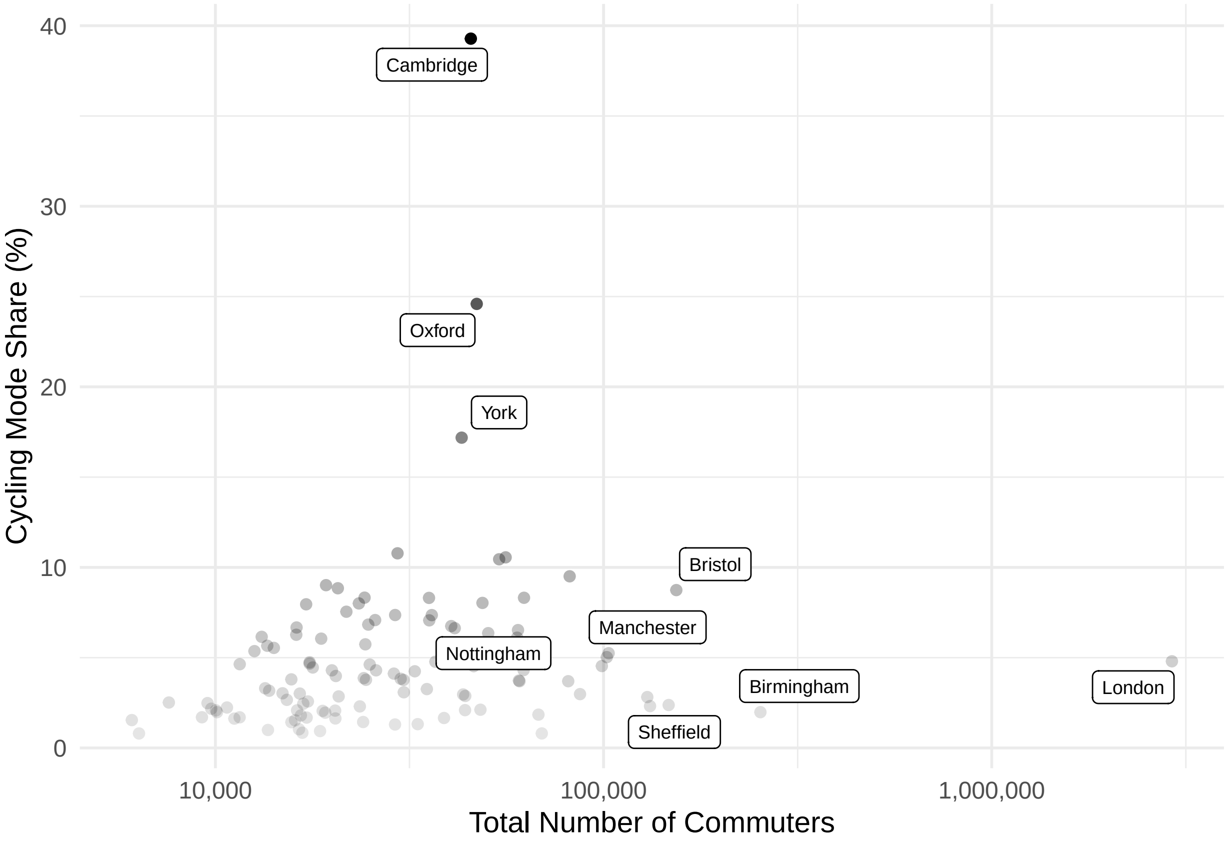
```
# number of MSOAs in each city
no_msoas <- city_names %>% dplyr::group_by(city) %>% dplyr::tally()

##### CHOOSE YOU CITY
chosen_city <- "Manchester"
#create a directory to store data related to this city (does nothing if directory already e
dir.create(paste0("../data/", chosen_city), showWarnings = FALSE)
# create sub-directory to save plots as well
dir.create(paste0("../data/", chosen_city,"/Plots"), showWarnings = FALSE)

##### CHOOSE YOU CITY

# flow data from the 2011 census https://www.nomisweb.co.uk/census/2011/bulk/rOD1
flows <- read_csv('../data-raw/flow_data.csv')
```

```
## Parsed with column specification:
## cols(
##   ‘Area of residence‘ = col_character(),
##   ‘Area of workplace‘ = col_character(),
##   ‘All categories: Method of travel to work‘ = col_double(),
##   ‘Work mainly at or from home‘ = col_double(),
##   ‘Underground, metro, light rail, tram‘ = col_double(),
##   Train = col_double(),
##   ‘Bus, minibus or coach‘ = col_double(),
##   Taxi = col_double(),
##   ‘Motorcycle, scooter or moped‘ = col_double(),
##   ‘Driving a car or van‘ = col_double(),
##   ‘Passenger in a car or van‘ = col_double(),
##   Bicycle = col_double(),
##   ‘On foot‘ = col_double(),
##   ‘Other method of travel to work‘ = col_double()
## )
```

```
###############
# MERGING NAMES WITH FLOW DATA (TO GET INTERNAL FLOWS IN ANY CITY)
###############

# add a column with the city name corresponding to each Residence MSOA
flows <- flows %>% left_join(city_names[,c("MSOA11CD", "city")],
                             by = c("Area of residence" = "MSOA11CD")) %>%
  rename(city_origin = city) # rename column so that we know it is referring to the 'Area o

# add a column with the city name corresponding to each Workplace MSOA
flows <- flows %>% left_join(city_names[,c("MSOA11CD", "city")],
                             by = c("Area of workplace" = "MSOA11CD")) %>%
  rename(city_dest = city) # rename column so that we know it is referring to the 'Area of

# get mode share for all cities - just for report
cycle_mode_share <- flows %>%
  filter(city_origin == city_dest) %>%   # only internal flows
  group_by(city_origin, city_dest) %>%
  summarize(mode_share = (sum(Bicycle) / sum(`All categories: Method of travel to work`)) *
            all = sum(`All categories: Method of travel to work`)) %>%
  arrange(mode_share)
```

```
## `summarise()` regrouping output by 'city_origin' (override with `.groups` argument)
```

```
# plot
cycle_mode_share %>% #filter(city_origin != 'London') %>%
  ggplot(aes(all, mode_share)) +
  geom_point(aes(alpha = mode_share), show.legend = FALSE) +
  # some filtering labels for aesthetic purposes. Add some cities explicitly as they are in
  geom_label_repel(aes(label = ifelse(mode_share>12 | all> (2.3*mean(all)) | city_origin %i
                   size =2.5) +
  labs(x="Total Number of Commuters", y = "Cycling Mode Share (%)") +
  scale_x_continuous(trans='log10', labels = scales::comma) +
  theme_minimal()
```



2. What Affects the Decision To Cycle

3. Planning Cycling Networks

4. Potential Cycling Demand

$$\begin{aligned} logit(C_p) = & -4.018 - 0.6369d + 1.988\sqrt{d} + 0.008775d^2 \\ & -0.2555s + 0.00206ds - 0.1234\sqrt{d}s \end{aligned} \quad (1)$$

5. Routing

6. Road Segment Prioritization

7. Analysis of Methodology

8. Conclusions

References