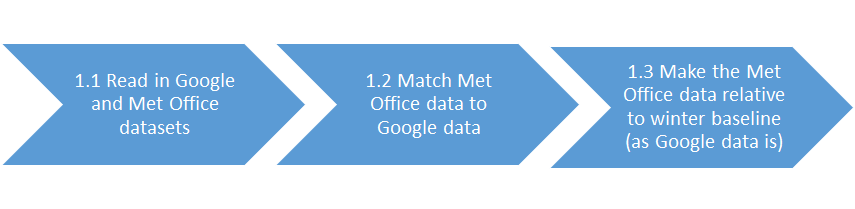
**Approach: Back-end**

Our back-end workflow is complicated, but has 3 broad stages, with 3 sub-stages:

1. Matching Met Office data to Google data
2. Matching Government (Natural England and OS Greenspace data) to Google data
3. Building a model and forecasting park busyness from it
4. **Matching Met Office data to Google data**

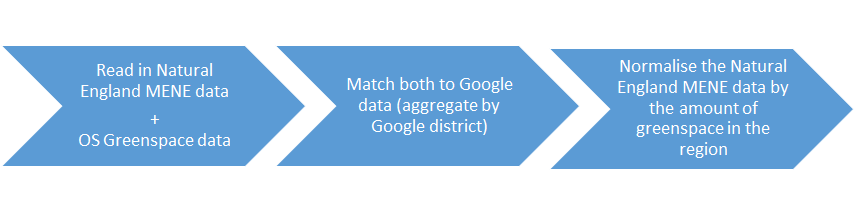
****

**1.1** Our first challenge was to create R code for reading in the open datasets made available by Google and the Met Office as part of their COVID-19 response. The Google data for England is downloaded and extracted from the global CSV file available at <https://www.google.com/covid19/mobility/> through the function **read.googlemobility.R.** The Met Office data for England is downloaded from the UK COVID reporting region weather data available at (<https://metdatasa.blob.core.windows.net/covid19-response/index.html>) through the function **read.metofficecovid.R**.

**1.2** The Google and Met Office data are slightly different. The Met Office data on daily weather conditions are broken down by COVID reporting region, whilst Google uses its own definitions of a region/district. We had to figure out how Google defined it’s regions (and produce a map of each region) and aggregate the weather from each of the COVID reporting regions (typically smaller than the Google regions) to each of the Google regions (**match.metoffice.R)**.

**1.3** Google’s data is in the form **% busyness relative to how busy parks were on the same day of the week (Mon-Sunday) in the 5‑week winter period Jan 3 – Feb 6, 2020.** Positive percentages mean that parks are more busy than the average (median value) during this period, negative percentages mean they are less busy (e.g. + 100% = 2 x as busy, 0% = as busy as winter, -100% 0 visitors). The Met Office data is in absolute values.  
  
Thus, we had to write some code to make the Met Office data relative to the weather in the Google baseline period; so that, for example, the temperature on a particular day of the week is represented as % increase relative to the average temperature on that day of the week in the winter baseline period. Our function **relative2baseline.R** does this.

1. **Matching Government (Natural England and OS Greenspace data) to Google data**

****

**2.1** In order to control for the the density of gardens and parks in each Google district, we read in Ordnance Survey (OS) data that had been pre-processed by the Office for National Statistics (ONS) in **match.OSgreenspace2google.R** (lines 1-30).

**2.2** We then tidied this data (lines 31-55) and merged data for parks and gardens together (lines 56-86). These data were only available at lower-tier local authority level so in order to aggregate them at the level of the Google district, we first read in an openly available lower-tier local authority to upper-tier local authority lookup table (lines 87-104), merged this with the OS data (lines 105-118), before manually matching the upper-tier local authorities to ogle districts (lines 119-154).

**2.3** Lastly, the OS data was aggregated at the Google district level (lines 155-186) using sums for frequencies of gardens in a Google district and areal coverage by gardens in a Google district and weighted means (i.e. weighted by numbers of addresses in a Google district) for all other metrics (average size of gardens in a district; average distance to nearest public greenspace, average count of public greenspace within 1000m of a typical address; and average combined size of all public greenspace within 1000m of a typical address).

**2.4** Our next task was to obtain ten years of survey data from Natural England’s Monitor of Engagement with the Natural Environment (MENE) survey. The first section of **getandmatch.mene.R** (lines 1-31) reads in MENE’s visit-based data file, where each row corresponds to a visit. Using the grossing-up weights provided by Natural England, we aggregated the data by upper-tier local authority and produced an estimated annual (i.e. divided by 10) number of visits to green spaces taken by the adult population of England (aged 16+) to each upper-tier local authority.

**2.5** Again we manually matched upper-tier local authorities to Google districts in **getandmatch.mene.R** (lines 32-63).

**2.6** Of course, districts with higher populations confer higher numbers of visits, so we decided to normalise visit estimates by the population of each google district by reading in corresponding respondent-based MENE data from Natural England (lines 64-69), estimating the adult population of each upper-tier local authority (lines 70-78), and aggregating at the Google district (lines 79-116).

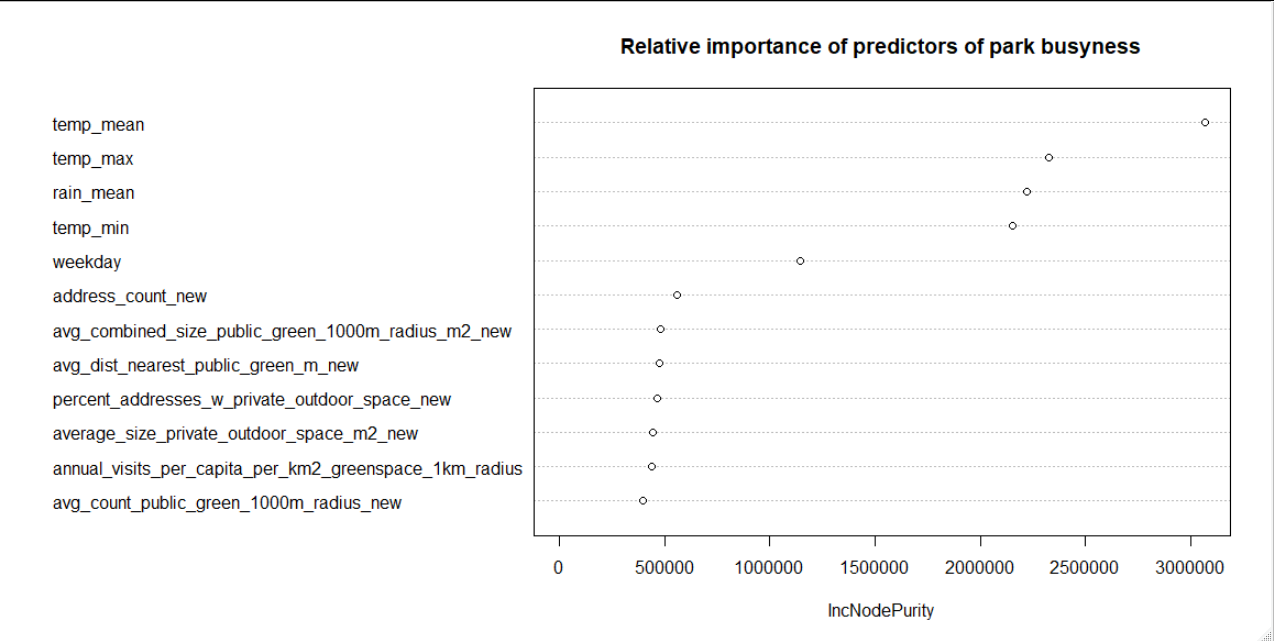
**2.7** As well as these per capita estimates, we further made provision for the normalisation of visit estimates per km2 of greenspace area within the Google district by reading in the OS data again (lines 117-132; see 2.3), merging with the MENE data (lines 133-136), and finally creating variables which represent: (a) raw visit estimates, (b) visit estimates relative to the population of each Google district, (c) visit estimates per km2 greenspace in the 1km surrounding a typical address in the Google district, and (d) visit estimates relative to both population (c) and greenspace access (d) (lines 137-143).

1. **Building a model and forecasting park busyness from it**

****

Finally, after matching all of the temporal (Met Office weather data) and single measure data (Natural England and OS Greenspace data) to the Google data, we used the combined dataset to model how the weather and social science data predicts the historical Google park busyness data **(create.model.R)**. We deploy a random forest regression model for this, including the explanatory variables/predictors (name in the model given in brackets):

1. **Mean temperature on the day (temp\_mean):** This is important because people are more likely to visit parks when it’s hotter! (Met Office data)
2. **Maximum temperature on the day (temp\_max):** This is important because people are more likely to visit parks when it’s hotter! (Met Office data)
3. **Mean rainfall on the day (rain\_mean)**: This is important because people are less likely to visit parks when it’s raining! We didn’t include maximum or minimum rainfall because minimum rainfall is not provided by the Metoffice dataset, and OpenWeathers’ dataset for rainfall is not discrete enough to calculate an accurate maximum rainfall that corresponds to the Metoffice values. (Met Office data)
4. **Minimum temperature on the day (temp\_min):** This is important because people are more likely to visit parks if it’s warmer in the evening. (Met Office data)
5. **The day of the week it is (weekday):** This is important because people are likely to behave differently on weekends than on weekdays (e.g. more of an increase because people have more free time
6. **The number of addresses within the Google district (address\_count\_new)** This is important as an estimate of population density in the district. A higher population could be associated with higher visits or conversely, a more urbanised area may have less accessible greenspace to visit. (Office for National Statistics/Ordinance Survey data)
7. **The average total size of all publicly accessible green spaces within 1km of a typical address within the Google district i.e. averaged across all addresses within the district (avg\_combined\_size\_public\_green\_1000m\_radius\_m2\_new)** This is important because it provides an estimate of the average areal coverage by accessible public green space for any given household in a region (Office for National Statistics/Ordinance Survey data)
8. **The average distance in metres from a typical address within the Google district to the nearest public greenspace i.e. averaged across all addresses within the district (avg\_dist\_nearest\_public\_green\_m\_new)** This indicates how readily accessible a public greenspace is for a typical household within the district (Office for National Statistics/Ordnance Survey data).
9. **The percentage of addresses within the Google district which had access to a private outdoor space e.g. garden, balcony, or yard (percent\_addresses\_w\_private\_outdoor\_space\_new)** This is important because if people have access to gardens they might substitute time in public greenspace for time in their own gardens. (Office for National Statistics/Ordinance Survey data)
10. **The average size of a private garden or outdoor space (e.g. yard, balcony) for a typical address within the Google district i.e. averaged across all addresses with a private outdoor space within the district (average\_size\_private\_outdoor\_space\_m2\_new).** Like (9), this speaks to potential substitutability - those with larger private outdoor spaces may substitute time in these for time in public greenspaces; those with smaller (e.g. shared or non-natural) private outdoor space, may choose to visit public greenspace more often (Office for National Staitstics/Ordnance Survey data).
11. **The number of recreational visits to green spaces within the Google district taken by the adult population of England (aged 16+) annually (averaged across 10 years of data) divided by the 16+ population of the Google district and the total size of all publicly accessible greenspace within 1000m of a typical address within the Google district (annual\_visits\_per\_capita\_per\_km2\_greenspace\_1km\_radius)** This is important because it provides an estimate of the number of visits to greenspaces in a region relative to the population size and the amount of available greenspace. (Natural EnglandMonitor of Engagement with the Natural Environment data)
12. **The average number of public green spaces within 1km of a typical address within the Google district i.e. averaged across all addresses within the district (avg\_count\_public\_green\_1000m\_radius\_new)** This is important because it provides an estimate of the average number of accessible parks for any given household in a region; more choice could mean more visits, or conversely more indecision. (Office for National Statistics/Ordinance Survey data)

After building this model, we use it to predict how busy parks will be over the coming days, based on these variables. The Natural England and OS Greenspace data for each district remain constant for this prediction/forecast, whilst the weather data is brought in from 6-day OpenWeather forecasts. A prediction of park busyness (% change from winter baseline) is then made and displayed in our app as a white bar in the bar graph. The bar graph is shown per weekday because Google baselines are generated per-weekday, and so a plot per weekday represents a true time series.

The variable importance plot (above) of our random forest model indicates that temperature was the most important predictor of Google park visit trends, followed closely by rain. Day of the week was also important - probably indicating that whether it’s a weekend effects park visits a lot! The social science data had a subtler effect on model estimates, but were nonetheless important

**Approach: Front-end**

The front-end is an RShiny app (**app.R**) to display the historical park visit data alongside the predictions of park usage for the coming days. This app has two main components:

* A bar graph of park busyness over time, relative to the winter baseline period. Less busy than winter is displayed in grey, more busy than winter is displayed in green. Users can interact with the graphs in the following ways:

1. View the whole historical dataset for every day where there is Google data available, with no forecast
2. Select the time period to view the data on the whole historical dataset on the graph and on the map, using the calendar on the sidebar.
3. View the historical datasets for each weekday (a true time series), with prediction for the next weekday displayed as a white bar.
4. Add a scaled percentage change plot for temperature and rainfall to both the historical dataset for each weekday and the whole historical dataset.

* A map with the boundaries of the Google districts and the busyness of parks (less busy than winter is displayed in darker shades of grey, more busy than winter is displayed in darker shades of green). Users can interact with this map by:

a) Selecting the time period to view the mapped-out data on the whole historical dataset (forecasts are not yet displayed on map), using the calendar on the sidebar.

b) Selecting a region on the sidebar, which is then highlighted with a purple border on the map.

The bar graph was produced by re-visualising the Google data in the more appropriate format of a bar-graph with negative and positive colours to make visualising how busy it is relative to winter more intuitive. In addition, we changed percentage change into the form of ‘how many times as busy is it’, as this is more intuitive. The function **plot.googlemobilitydistricts.R** does this for the whole time series, the function **plot.parkvisits.R** does this on a per-weekday basis that can be interacted with in the app.

The map was a tricky one; after identifying how the Google regions were defined, the challenge was to match these to shapefiles from Ordinance Survey open boundary line data (<https://www.ordnancesurvey.co.uk/business-government/products/boundaryline>) in the script **match.OSopenBoundary2google.R**. To our knowledge, this has not been done before and this is the first map of the Google regions, which we make freely available in the ‘spatial’ folder of our Open Science Framework data repository (<https://osf.io/c7kg4/>) - which includes all the other data that goes into the