



How to map a 20-minute neighbourhood:

Data and methods



CONTENTS

INTRODUCTION	3
20-MINUTE NEIGHBOURHOODS: WHAT, WHEN AND WHERE?	5
WHAT DATA SHOULD I INCLUDE IN MY ANALYSIS?	7
WHERE TO SOURCE YOUR ROADS AND STREETS NETWORK	10
HOW TO CREATE A HEXAGONAL GRID ACROSS YOUR LOCAL AUTHORITY AREA	14
HOW TO PREPARE YOUR FACILITIES LAYERS	15
HOW TO COMPLETE THE NETWORK ANALYSIS	17
OPTIONAL FURTHER ANALYSIS	28
APPENDIX	29



INTRODUCTION

This technical note provides step-by-step guidance on how to conduct a 20-minute neighbourhood network analysis using ArcGIS Pro. The analysis focuses on creating service areas around facilities commonly used in daily life and combining these service areas to assess the density of services within a given area. The final output is a map that highlights areas with varying levels of access to essential services within a 10-minute walking distance.

Policy Context

The National Planning Framework 4 (NPF4) has introduced policy requirements related to 20-minute neighbourhoods and the concept of local living. The overarching policy intent, as outlined in NPF4, is:

“To encourage, promote and facilitate the application of the Place Principle and create connected and compact neighbourhoods where people can meet the majority of their daily needs within a reasonable distance of their home, preferably by walking, wheeling or cycling or using sustainable transport options.” (p61, NPF4, Scottish Government)

Purpose of this guidance

To support the implementation of this policy, the Improvement Service has compiled best practices from local authorities across Scotland

that have already undertaken 20-minute neighbourhood analyses. This includes links to relevant data sources, practical advice, and recommendations for producing meaningful analytical outputs. The approach outlined in this guidance is based on the methodology developed by Highland Council, which serves as an example of best practice in assessing access to a diverse range of facilities and services in an integrated way.



This document is designed to help users understand and apply a consistent approach to 20-minute neighbourhood analysis. It includes:

- An explanation of what a 20-minute neighbourhood is.
- Guidance on when and where a 20-minute neighbourhood analysis should be conducted.
- Recommendations on the services and facilities to include in the analysis.
- Information on relevant data sources.
- Step-by-step instructions for conducting a network analysis and aggregating findings into a map.
- A discussion of the limitations of the proposed approach.
- Suggestions on additional analysis that could be undertaken.



Caveats

The methodology outlined in this guidance has several limitations and considerations that should be noted:

1. Scope of Considerations

There are an almost limitless number of factors to consider when setting out to complete a 20-minute neighbourhood analysis. This guidance focuses on a core set of considerations that can be applied nationally. However, softer factors, such as the perceived sense of safety within a neighbourhood, are not included due to the lack of available national data.

2. Catchment Area Challenges

Many services, such as schools or health facilities, typically serve much wider catchment areas than a 10-minute walk. As a result, it is unlikely that most neighbourhoods will have perfect access to all types of facilities.

3. Customisation by Local Authorities

This guidance is intended as a starting point for calculating and assessing 20-minute neighbourhoods. Local authorities are encouraged to adapt or expand upon this approach to better suit their specific local contexts and priorities.

4. Temporal Limitations

This analysis does not account for the temporal availability of services and facilities. For example, it does not consider whether a shop has limited opening hours, or if a health facility is closed on certain days.

5. Public Transport Accessibility

While this guidance includes access to bus stops and railway stations, it does not cover public transport accessibility in detail. Factors such as the frequency, reliability, or scheduling of buses and trains are not incorporated into the analysis, nor are public transport routes.

6. Software and Licensing Requirements

Completing this analysis requires paid Esri software and licenses, including ArcGIS Pro, Network Analyst, Data Interoperability, and the UK Data Loader. Users should ensure they have access to these tools before starting the analysis.

20-MINUTE NEIGHBOURHOODS: WHAT, WHEN AND WHERE?

What?

The Scottish Government defines a 20-minute neighbourhood as a concept that “aims to provide access to the majority of daily needs within a 20-minute walk, wheel, or cycle from home.”¹ While straightforward in theory, the reality is more complex, as noted by Dr. Al Waer of Dundee University: “a 20-minute neighbourhood defies clear definition.”²

At its core, a 20-minute neighbourhood is generally considered a place where facilities essential for daily life can be accessed within a 10-minute walk or cycle from home (10 minutes to a facility and 10 minutes back). However, the specific facilities required, and the feasibility of accessing them, may vary significantly depending on individual needs and abilities. Different authorities may tailor their analyses to reflect the unique needs of their communities, amending the list of facilities assessed.

When is a 20-minute neighbourhood analysis useful?

Access to facilities and services, as well as the development of new homes, is constantly evolving. Conducting a thorough analysis can help identify areas that currently lack services

and highlight undeveloped areas with strong potential for accessibility should they be developed.

To ensure decisions are informed by up-to-date information, it is recommended that this analysis be repeated on an annual basis. Designing the methodology to be replicable as data is updated will help streamline future iterations and maintain consistency over time.

Where should a 20-minute neighbourhood analysis be conducted?

The concept of 20-minute neighbourhoods is typically urban-focused and may not be suitable for all contexts, particularly rural areas with low or sparse populations. As the Scottish Government notes:

“The 20-minute neighbourhood concept aims to provide access to the majority of daily needs within a 20-minute walk, wheel, or cycle from home. It is an approach likely to be more readily achievable in urban places, towns, villages, and cities. It is designed to be applied flexibly, in response to local circumstances.”³



- ¹ [Scottish Government Planning Guidance: Local living and 20 minute neighbourhoods](#), 2024
- ² [New guide on how to create 20-minute neighbourhood launched](#), 2024
- ³ [Scottish Government Planning Guidance: Local living and 20 minute neighbourhoods](#), 2024

For rural communities, with their dispersed populations and vast geographic areas, the analysis may have limited value. A practical starting point for identifying suitable locations is to focus on settlements with a population of 500 or more, as identified by the [Population Estimates for Settlements and Localities in Scotland](#). This threshold can be adjusted to suit specific contexts; for example, focusing on settlements with populations of 1,000 or more may be more appropriate in certain areas.

Local development plans (LDPs) can also be a valuable resource for narrowing down areas of focus. For instance, Highland Council conducted their analysis by prioritising areas defined in their LDP as “Growing Settlements” and “Settlement Development Areas.” Authorities may wish to adopt a similar approach, tailoring it to their specific policy priorities and local needs.



Why are 20-minute neighbourhoods important?

A 20-minute neighbourhood allows people to meet their daily needs within a short walk or cycle from their home. Designing our towns and cities with accessibility in mind has the following benefits:

Health Benefits: Encouraging walking, cycling, and wheeling helps people incorporate physical activity into their daily routines, promoting healthier lifestyles.

Environmental Benefits: Reducing reliance on cars decreases traffic congestion and improves air quality, contributing to a more sustainable future.

Economic Benefits: Local shops and businesses benefit from increased footfall, supporting the local economy.

Social Benefits: By fostering more frequent interactions between neighbours, 20-minute neighbourhoods can strengthen community bonds and improve overall wellbeing.

As noted by the [Town and Country Planning Association](#), 20-minute neighbourhoods can enhance the liveability and sustainability of communities, creating places where people feel connected and supported.

WHAT DATA SHOULD I INCLUDE IN MY ANALYSIS?

A 20-minute neighbourhood analysis requires a combination of spatial datasets and information about local facilities. The core dataset you need are a network of roads and streets and a point layer of various facilities.

Seek community advice

While we have provided suggested facilities you may wish to analyse (Table 1), communities understand their needs best and it is highly recommended that you survey members of your local community to understand their needs and priorities. Consider the following steps:

- **Identify Daily Needs:** Ask residents which services and facilities they rely on most frequently in their daily lives.
- **Tailor Your Analysis:** Weight your analysis towards the facilities that are most important to the community. For example, if healthcare and public transport are of greater importance in your area, these should be emphasized in the analysis.


By incorporating local views, your analysis will provide a more accurate and meaningful understanding of accessibility in your area.

Suggested facilities and weightings: Within Table 1, we have provided a suggested list of facilities that you may wish to include in your analysis, along with potential weightings⁴ to reflect their importance. This list, along with examples of facilities assessed by Highland Council and West Dunbartonshire, can be found in the appendix. You are encouraged to adapt or expand this list according to your local context.

Group facilities within your analysis: To simplify the analysis and interpretation of results, you may wish to group certain types of facilities together. For example:

- **Public Transport:** Combine bus stops, railway stations, and other transportation hubs into a single category.
- **Religious Facilities:** Group different places of worship together under one category if relevant to your community context.

Grouping facilities can make it easier to compare accessibility across broad service types while still capturing key trends.



The existing 20-minute neighbourhood policy in NPF4 (p61) refers to the following facilities:

- sustainable modes of transport including local public transport and safe, high-quality walking, wheeling and cycling networks;
- employment;
- shopping;
- health and social care facilities;
- childcare, schools and lifelong learning opportunities;
- playgrounds and informal play opportunities, parks, green streets and spaces, community gardens, opportunities for food growth and allotments, sport and recreation facilities;
- publicly accessible toilets;
- affordable and accessible housing options, ability to age in place and housing diversity.

⁴ Weighting in this case means giving different facilities a higher or lower score to reflect their relative importance to a 20-minute neighbourhood.

Table 1: 20-minute neighbourhood suggested facilities

Theme	Facility Type	Grouped Facilities	Suggested Weighting	Data source
Education	Nursery		0.8	OS NGD (land use features, site, description = “Children’s Nursery”)
Education	Primary School		1	Scottish Government
Education	Secondary School		0.8	Scottish Government
Shopping	Supermarket		1	3rd Party Data
Shopping	Local shop (e.g. Spar)		1	LA Gazetteer
Health	GP		1	Spatial Hub
Health	Pharmacy		1	Spatial Hub
Transport	Bus stop	Assess together as public transport access points, each scored 0.5 up to a maximum of 1	1	NaPTAN Or Corporate Address Gazetteer
Transport	Train station			
Transport	Ferry			
Play	Children’s play area		0.8	OS NGD (land use features, site, description = “Play Area”)
Environment	Park/Open Space		1	OS Open Greenspace
Sport Facilities	Golf course	Assess these facilities together as sports facilities, each scored at 0.4 up to a maximum of 1	1	Spatial Hub
Sport Facilities	Tennis court			
Sport Facilities	Bowling green			
Sport Facilities	Swimming pool			
Sport Facilities	Football/rugby/shinty pitch			
Sport Facilities	Leisure Centre			
Community Facilities	Library	Assess these facilities together as community facilities, each scored at 0.8, up to a maximum of 1	1	Spatial Hub
Community Facilities	Community facilities			OS NGD (land use features, site, description = “Community Meeting Place” or “Community Services”)
Community Facilities	Allotment			OS NGD (land use features, site, description = “Allotments”)
Financial Services	ATM		1	Corporate Address Gazetteer
Financial Services	Bank/Building Society		1	Corporate Address Gazetteer

Theme	Facility Type	Grouped Facilities	Suggested Weighting	Data source
Food and Drink	Pub	Assess these facilities together as food and drink, each scored at 0.3 up to a maximum of 1	1	OS NGD (land use features, site, description = “Public House”)
Food and Drink	Restaurant			Corporate Address Gazetteer
Food and Drink	Café			Corporate Address Gazetteer
Food and Drink	Takeaway			Corporate Address Gazetteer
Cultural Facilities	Museum	Assess these facilities together as cultural facilities, each scored at 0.3, up to a maximum of 0.6	0.6	OS NGD (land use features, site, description = “Museum”)
Cultural Facilities	Theatre			OS NGD (land use features, site, description = “Theatre”)
Cultural Facilities	Cinema			OS NGD (land use features, site, description = “Cinema”)
		Total possible score:	15.2	

The facilities identified in Table 1 can be drawn from a few different data sources. The data sources are provided as a suggestion only. Local authorities will generally have more comprehensive data, though it may require geocoding. Using the best possible data available will strengthen the analysis.

WHERE TO SOURCE YOUR ROADS AND STREETS NETWORK

You can create a network using Ordnance Survey's (OS) [National Geographic Database](#) (NGD) tool. All planning authorities and Public Sector Geospatial Agreement (PSGA) users can access this data. Please reach out to the Improvement Service or to OS if you are unsure who your NGD focal point is.

Step 1: To access the data, you must first login to the [NGD](#).

Step 2: Navigate to the OS [Select+Build](#) tool. You can use this tool to select from OS which road and street data you want to download.

Step 3: Choose “create a new recipe” from the top right of the page (Figure 1: Select+Build Recipe Library):

Figure 1: Select+Build Recipe Library

The screenshot shows the 'OS Select+Build Recipe Library' page. At the top right is a yellow button with a plus icon and the text 'Create a new recipe'. Below the header, there is a search bar labeled 'Find recipes' with the placeholder text 'Search by recipe name, author, description or content'. Below the search bar are two date selection fields: 'From creation date (dd/mm/yyyy)' with the value '04/11/2022' and 'To creation date (dd/mm/yyyy)' with the value '13/02/2025'. Both fields have calendar icons.

Step 4: Give your recipe a name, for example “Transport Network” and then select the data you want. We suggest you select the following:

Select all elements from the transport NGD (RAMI, Transport Features and Transport Network), give the recipe a name and choose create recipe (see Figure 2: Create your recipe for more details).

Figure 2: Create your recipe

The screenshot shows the 'Create your recipe' form. At the top is the title 'Create your recipe'. Below it is a text input field for 'Recipe name' with the value 'Transport Network'. Below the input field is a link 'Add a description'. Below that is a section titled 'Themes' with a search bar 'Search by feature type' and a 'Reset selection' link. Below the search bar is a list of themes with checkboxes and expand/collapse arrows: 'Address', 'Administrative and Statistical Units', 'Buildings', 'Geographical Names', 'Land', 'Land Use', 'Structures', 'Transport' (checked), 'RAMI' (checked), 'Transport Features' (checked), and 'Transport Network' (checked). At the bottom are two buttons: 'Cancel' and 'Create recipe'.

Once you create the recipe, you need to add a “data package”. Click the “add data package” button (Figure 3: Add data package)⁵

Figure 3: Add data package

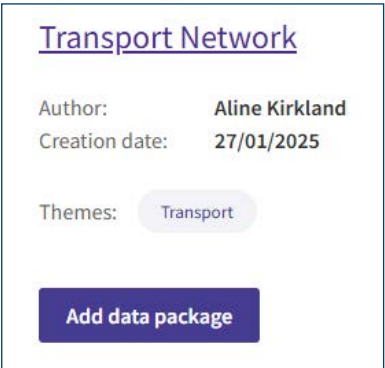
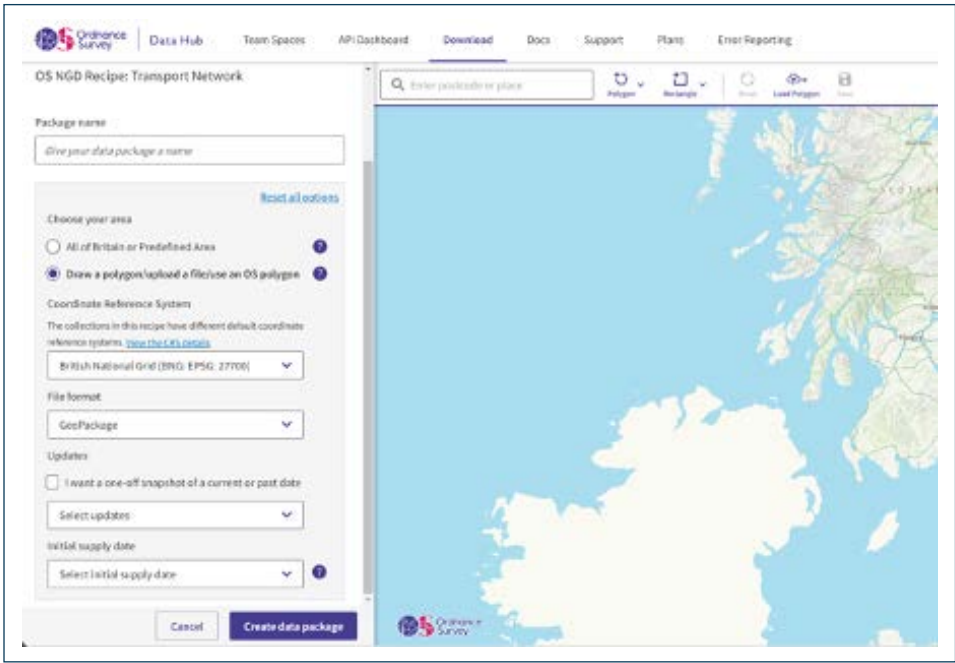


Figure 4: Create your data package



⁵ Please note, if “add data package” option is not available to you, please contact OS or your NGD admin focal point.

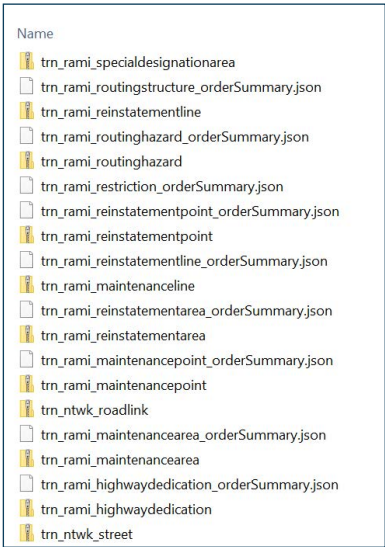
Give the package a name, select “draw a polygon/upload a file/use an OS polygon. You can use a Scotland Polygon to download the data or draw a polygon outline around your area of interest.

The reference system should be **British National Grid**, and the file format should be **GeoPackage**.

Once you have outlined your area of interest, select “**create data package**”. You will then receive an automated email from OS once your data is ready to download.

Step 5: Once you receive this email, download the network. There will be a mixture of JSON and zip files (Figure 5: Download of OS data package). You can delete the JSON files.

Figure 5: Download of OS data package



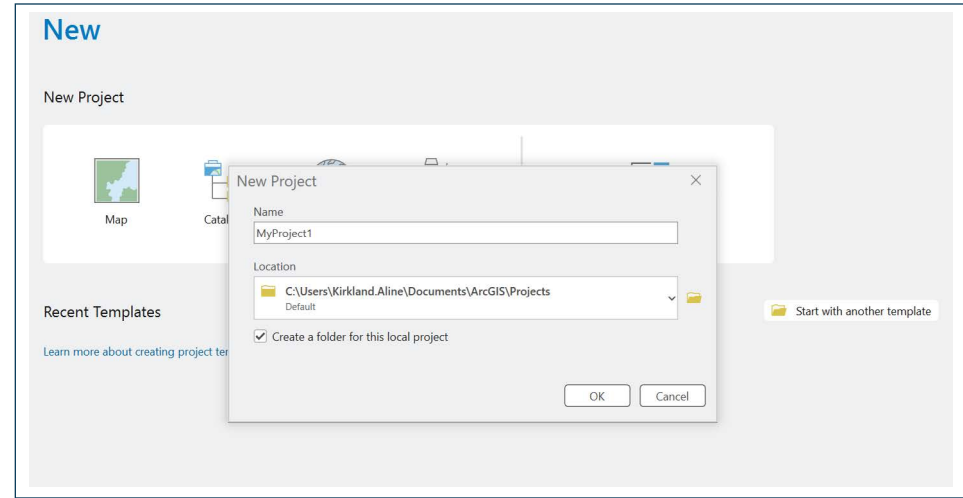
Step 6: To use this road and street data, you need to use the ArcGIS Pro Data Interoperability Tool and the [UK Data Loader](#) to transform the dataset into a working network. These are both paid for Esri toolboxes that your authority may need to purchase to process the network data into an analysis

ready dataset that can be loaded into the Network Analyst geoprocessing tool.

Once you have purchased the Data Interoperability toolbox and have purchased and downloaded the UK Data Loader,⁶ you need to add the UK Data Loader to your project to process the data.

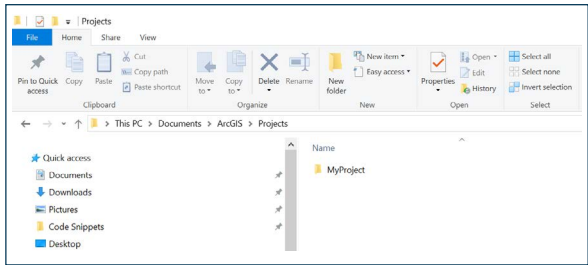
Step 7: Create a new, blank ArcGIS Pro project, by clicking on “Map”. Then give your project a name and set the project location/filepath (Figure 6: Setting up a new ArcGIS Pro Project):

Figure 6: Setting up a new ArcGIS Pro Project



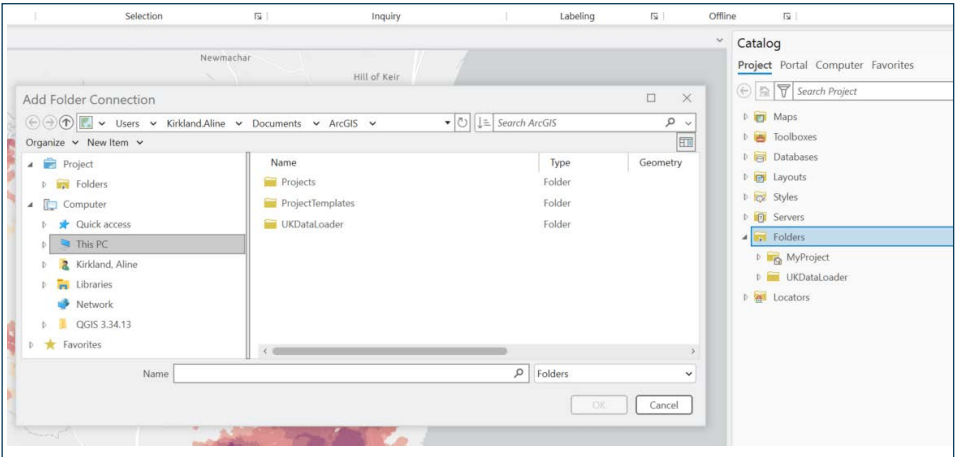
Step 8: Copy the UK Data Loader downloaded zip file to the folder that you created when you set up your project (Figure 7: New ArcGIS Pro Project Folder).

Figure 7: New ArcGIS Pro Project Folder



Step 9: Add the UK data loader to your project. You need to open the “Catalog” menu, right click on Folders and “Add a Folder Connection” (Figure 8: Adding UK Data Loader Package). Navigate to the location where your UK Data Loader files are saved (you may need to refresh the folder to see the data). Select the UK Data Loader folder and click “OK”.

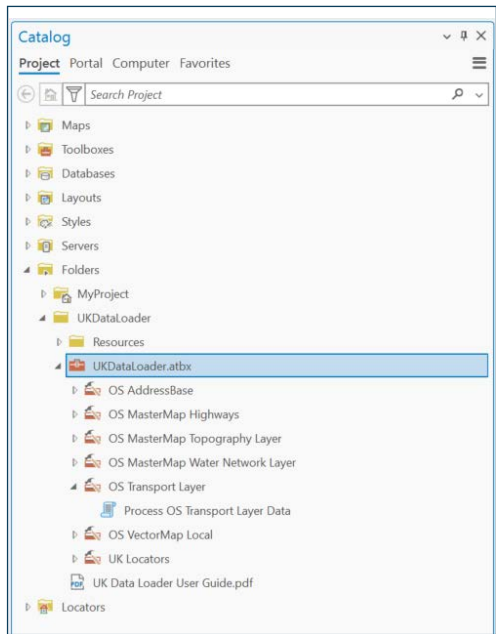
Figure 8: Adding UK Data Loader Package



6 UK Data Loader download: <https://www.arcgis.com/home/item.html?id=27d4e1a3170c428980e0f903d834779c>

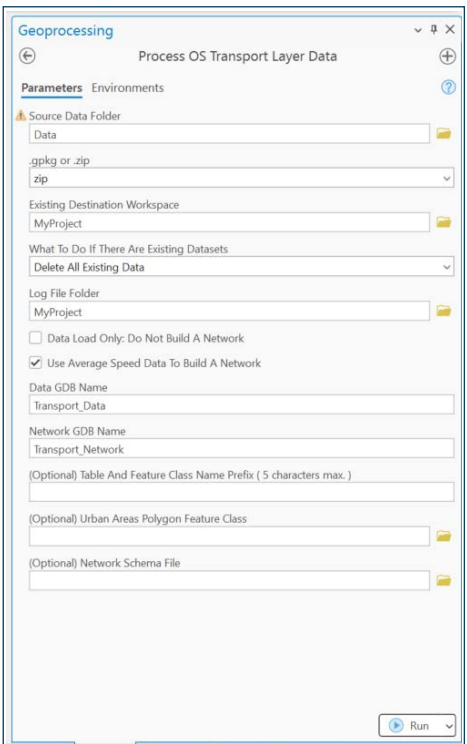
The UK Data Loader toolbox will now be available to use in your project (Figure 9: Accessing UK Data Loader Package):

Figure 9: Accessing UK Data Loader Package



Step 10: Click on the UKDataLoader.atbx file in your catalog, select the “OS Transport Layer” and then double click on “Process OS Transport Layer Data”. The geoprocessing window will then open (Figure 10: Processing the OS Transport Data Layer). Use the setting shown in Figure 10. Once you are happy with the settings you have selected, click “run”. The network may take upwards of a few hours to process so you may wish to do this step overnight.

Figure 10: Processing the OS Transport Data Layer



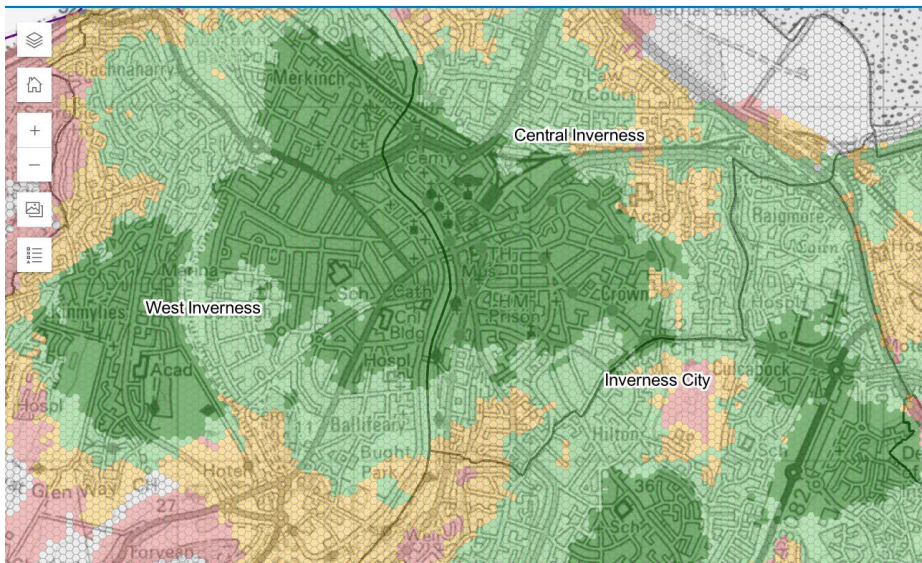
Once the process finishes, your network is ready to use with the ArcGIS Pro Network Analyst tool!

HOW TO CREATE A HEXAGONAL GRID ACROSS YOUR LOCAL AUTHORITY AREA

To measure access to facilities across your local authority area, covering existing households, and future housing and development sites, we recommend you draw on the approach taken by [Highland Council](#) and use a hexagonal grid (Figure 11: Highland Council Local Living Tool):

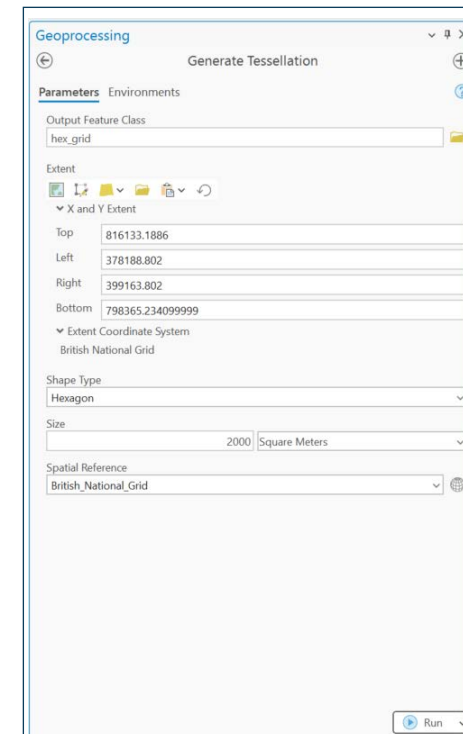
Figure 11: Highland Council Local Living Tool

Local Living Mapping



You can use the Data Management Tool “Generate Tessellation” in ArcGIS Pro to create hexagonal grid (Figure 12: Generate tessellation, creating a hexagonal grid). An area of 2,000 sqm per hexagon will create hexagons just over 55m wide. For the extent of your hexagonal grid, use the boundary of your local authority.

Figure 12: Generate tessellation, creating a hexagonal grid



HOW TO PREPARE YOUR FACILITIES LAYERS

Now that you have your network and a grid to work with, you need to gather your facilities datasets. You can either use datasets held by your own authority or make use of the datasets identified in Table 1.

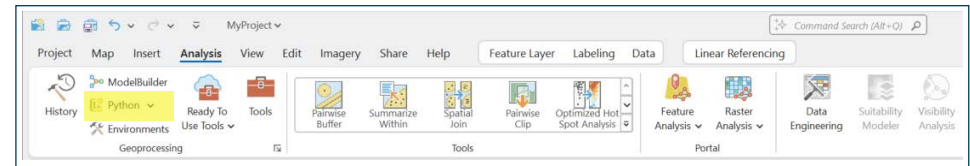
For this simplified example, I downloaded the following datasets from the Spatial Hub, NaPTAN, Scottish Government and OS:

- Sports facilities (Spatial Hub)
- Pharmacies (Spatial Hub)
- GP Practices (Spatial Hub)
- Retail points (GEOLYTIX)
- Public transport stops (NaPTAN)
- Play parks (OS)
- Primary and secondary schools (Scottish Government)

Step 1: Create a “Points of Interest” folder in your ArcGIS Pro project to store your downloaded facilities datasets and copy the datasets here. Some of the datasets may be in .csv or geopackage format so may require some processing when loading into ArcGIS Pro. All your facilities must be in the form of points, not polygons.

Step 2: Add a field to each of your points of interest called “Name” and assign to this field the layer name for each layer. For example, your GP layer should have a field called Name that contains the value “GP”. You can do this manually, or you can use the following ArcPy script, pasted into the Python window (ArcPy Script 1: Adding a name field to facilities). This script will target all point layers in your project and will copy their layer name and assign it to the Name field:

Figure 13: Accessing the Python window



ArcPy Script 1 is shown on page 16, however it is also saved as a .txt file as part of the 20-minute neighbourhood guidance package. To preserve the correct indenting of this script, and all others mentioned in the document, we strongly recommend that you take the scripts from the .txt files, rather than copying them from the document.

ArcPy Script 1: Adding a name field to facilities

```
import arcpy
import arcpy.mp
# Get the current ArcGIS Pro project
aprx = arcpy.mp.ArcGISProject("CURRENT")

# Loop through all maps in the project
for m in aprx.listMaps():
    for lyr in m.listLayers():
        if lyr.isFeatureLayer: # Ensure it's a feature layer
            desc = arcpy.Describe(lyr)
            if desc.shapeType == "Point": # Only process point
layers
                field_name = "Name"

                # Check if the "Name" field exists
                fields = [f.name for f in arcpy.ListFields(lyr)]
                if field_name not in fields:
                    arcpy.AddField_management(lyr, field_name,
"TEXT", field_length=100)

                # Update the "Name" field with the layer name
                with arcpy.da.UpdateCursor(lyr, [field_name]) as
cursor:
                    for row in cursor:
                        row[0] = lyr.name
                        cursor.updateRow(row)

print("Process completed: 'Name' field added and populated for
all point layers.")
```

Step 3: Merge together all of your facility point layers into one dataset. You can either use the “merge” tool in the geoprocessing window, or the following Python code to merge all the point features in your current project:

ArcPy Script 2: Merge together facility datasets

```
import arcpy
import arcpy.mp
import os

# Define output geodatabase and merged feature class name
output_gdb = r"C:\Users\Kirkland.Aline\Documents\ArcGIS\
Projects\MyProject\MyProject.gdb" # Change to your own
geodatabase path
merged_fc_name = "Facility_Points"
merged_fc_path = os.path.join(output_gdb, merged_fc_name)

# Get the current ArcGIS Pro project
aprx = arcpy.mp.ArcGISProject("CURRENT")

# List to store point feature layer paths
point_layers = []

# Loop through all maps in the project
for m in aprx.listMaps():
    for lyr in m.listLayers():
        if lyr.isFeatureLayer: # Ensure it's a feature layer
            desc = arcpy.Describe(lyr)
            if desc.shapeType == "Point": # Only process point
layers
                point_layers.append(lyr.dataSource)

# Merge point layers into a single feature class
if point_layers:
    arcpy.Merge_management(point_layers, merged_fc_path)
    print(f"Successfully merged {len(point_layers)} point layers
into: {merged_fc_path}")
else:
    print("No point layers found to merge.")
```

The resulting file (Facility_Points) should automatically appear in your project. The attribute table will contain a field called “Name” that contains the type of facility each point represents.

Your data is now ready to be used in the Network Analyst tool!

HOW TO COMPLETE THE NETWORK ANALYSIS

There are various types of network analysis. For 20-minute neighbourhoods calculating service areas is the most appropriate. A service area is the distance you can travel from a set facility. Service areas are used to answer questions like; “how much of the neighbourhood is within a 10-minute walk of my shop?”.

To assess the existence of 20-minute neighbourhoods within your local authority area, you need to calculate service areas for your Facility_Points layer. You can do this using the ArcGIS Pro Network Analyst tool.

The inputs for this analysis are:

- The OS NGD network you created.
- The facility layers you created (Facility_Points)

The outputs from this analysis are:

- A series of polygon layers showing the service areas for each facility type.

Step 1: Click on the Network Analysis tool under the analysis tab on the ribbon at the top of ArcGIS Pro. Then click “Data Source” and select the network you created earlier to use to calculate your service areas (Figure 14: Selecting your transport network):

Figure 14: Accessing Network Analyst

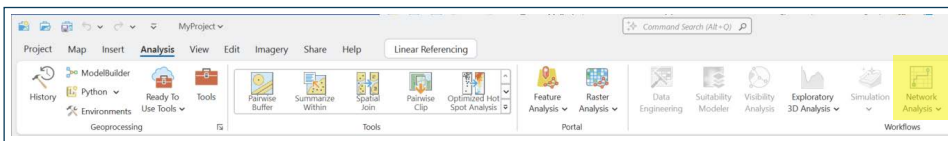
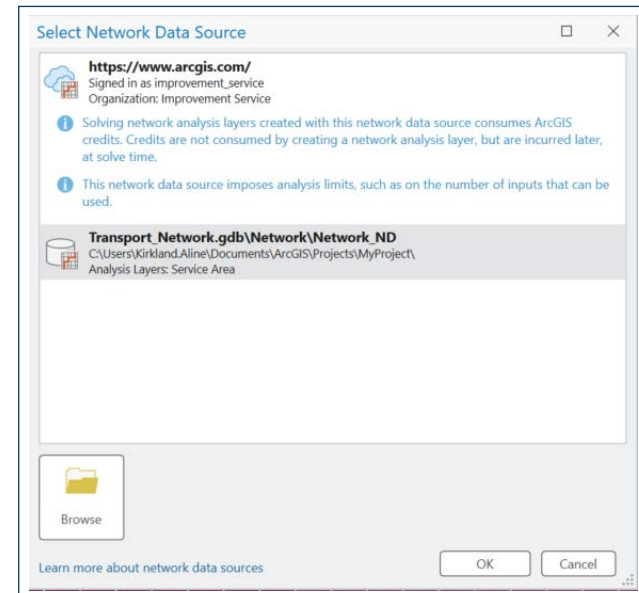
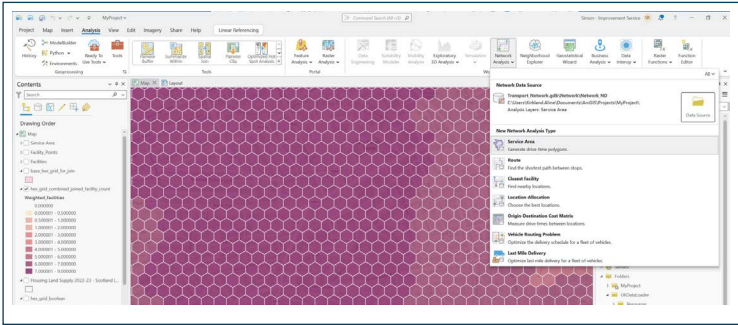


Figure 15: Selecting your transport network



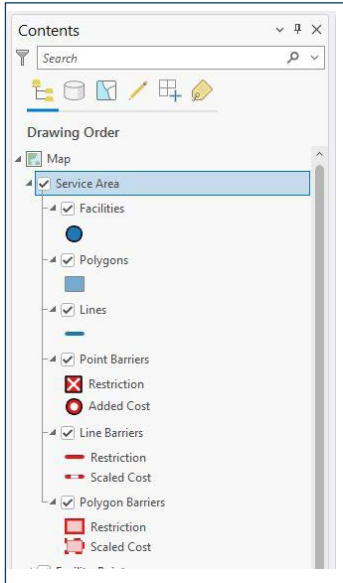
Once your network is selected, click on “Network Analysis” again and choose “Service Area” under “New Network Analysis Type” (Figure 16: Selecting Service Areas).

Figure 16: Selecting Service Areas



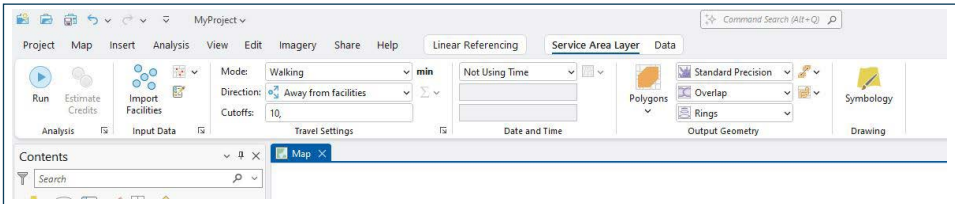
You will then see a Service Area layer has appeared in your contents (Figure 17: Network Service Area):

Figure 17: Network Service Area



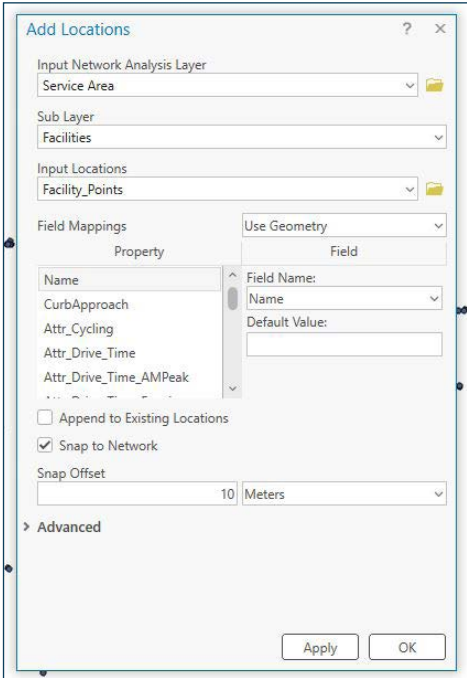
Step 2: Click on the “Service Area Layer” option that will now have appeared on the ribbon (Figure 18). You can now “Import Facilities”

Figure 18: Importing facilities into network



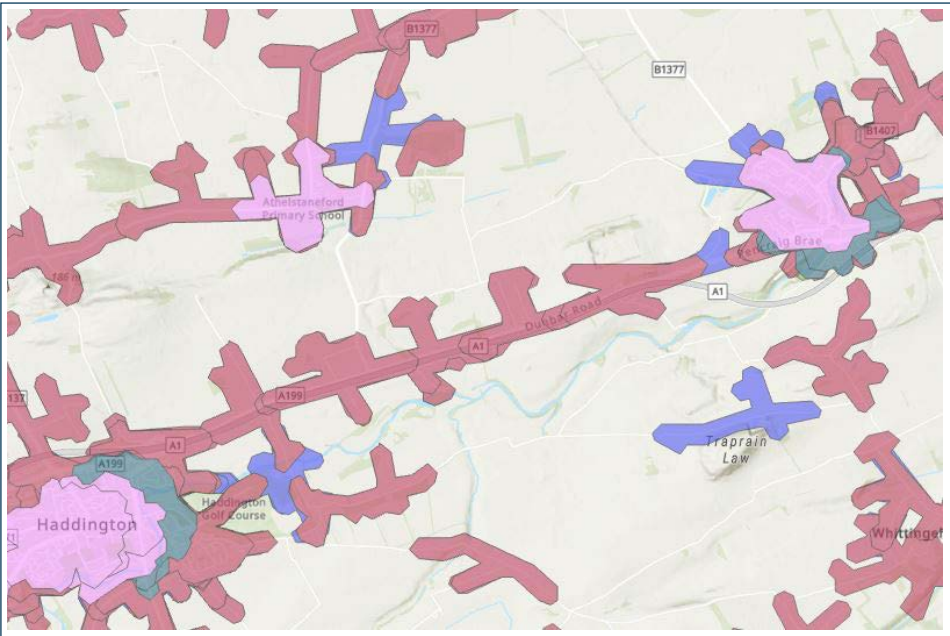
The following window will then open (Figure 19). Below are suggested settings. It is important you use the “Name” field you created earlier so you can keep track of your facility types through the analysis:

Figure 19: Processing service area



Once you are happy with your settings, click ‘OK’ and a progress bar will appear. Once the progress bar completes, your facilities should be imported into the Service Area. Next select “Run” from the ribbon at the top of the screen, this will run the process to create service areas. Your output once this process completes, should look similar to Figure 20.

Figure 20: Output from service area analysis



The service area layer will be called “Polygons” and will be visible in the Service Area group in your contents panel. Export this layer to your geodatabase and give it the name “Facility_Polys”.

The attribute table in Facility_Polys (Figure 21) should contain a “Name” field that outlines what each service area covers (is it the service area for a school or a GP etc):

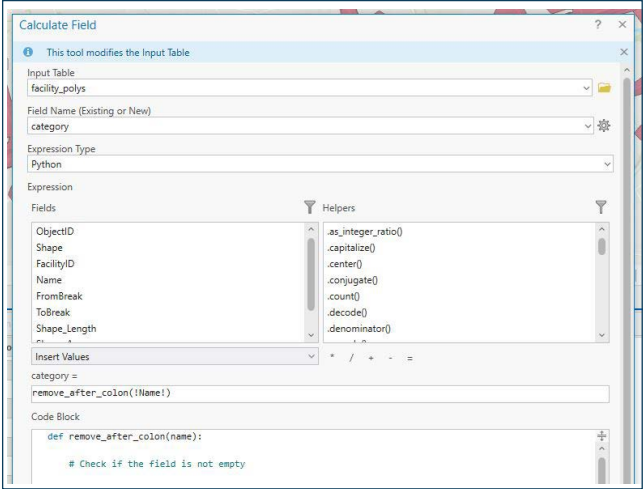
Figure 21: Output service area attribute table

A screenshot of the 'facility_polys' attribute table. The table has the following columns: ObjectID, Shape, FacilityID, Name, FromBreak, ToBreak, Shape_Length, and Shape_Area. The data rows show three polygons with their respective attributes.

ObjectID	Shape	FacilityID	Name	FromBreak	ToBreak	Shape_Length	Shape_Area
114	Polygon	4305	eastlothian_school_data1: 0 - 10	0	10	5986.011156	1024375
436	Polygon	4288	eastlothian_school_data1: 0 - 10	0	10	5093.331832	1381875
444	Polygon	4306	eastlothian_school_data1: 0 - 10	0	10	5561.765234	1434375

Step 3: The name field will need to be transformed in the output, trimming it so it is more useable for analysis. You can create a new field called “Category” trimming out unnecessary characters from the Name field by clicking “Calculate” at the top of your attribute table and using the following settings (Figure 22) and Python code.

Figure 22: Using calculate field to remove unwanted values from attribute table



The code is available below:

```
remove_after_colon(!Name!)
def remove_after_colon(name):
    # Check if the field is not empty
    if name:
        # If a colon exists, split and return the first part;
        # otherwise return the original value
        return name.split(":")[0].strip() if ":" in name else
        name.strip()
    else:
        return ""
```

Step 4: You now need to join the service area polygons to the grid you created earlier. The ideal output is a hex_grid layer with an attribute table that contains a 0 if no facilities are present within the grid cell, or a 1 for each facility type that is present in the grid.

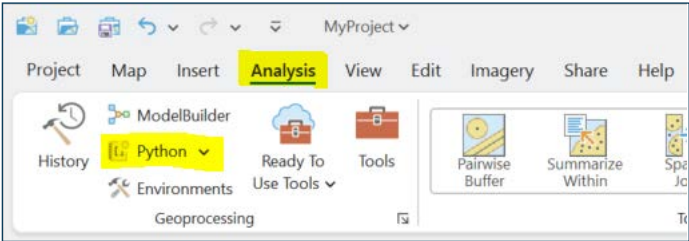
The output will look like the image below:

Figure 23: Joined hex_grid output attribute table

BJECTID	geom	geom_Length	geom_Area	Stops_NaPTAN	Retail_points	Pharmacy	Primary_schools	Play_Parks	GP	Sports_facility	Secondary_schools	Facility_Count
1	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
2	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
3	Polygon	173.20514	2165.065	1	0	0	0	0	0	0	0	1
4	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
5	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
6	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
7	Polygon	173.20504	2165.0625	1	0	0	0	0	0	0	0	1
8	Polygon	173.20504	2165.0625	1	0	0	0	0	0	0	0	1
9	Polygon	173.20514	2165.065	1	0	0	0	0	0	0	0	1
10	Polygon	173.20514	2165.065	1	0	0	0	0	0	0	0	1
11	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
12	Polygon	173.20514	2165.065	1	0	0	0	0	0	1	0	2

You can achieve this by manually applying a definition query to each category type in the Facility_Polys, and then spatially joining each category type to the grid. However, this analysis can be more easily achieved using ArcPy, pasted into the Python window. A code snippet is shared below. This script is also available as a .txt file within the 20-minute neighbourhood guidance package. We strongly recommend you copy the code from the .txt file. You will need to change the filepath (as highlighted in ArcPy Script 3 below) to your own project geodatabase and check that your layer names match the below code (hex_grid and Facility_Polys):

Figure 24: Opening the Python window



ArcPy Script 3: Spatial join facility service areas to a hexagonal grid

```
import arcpy
import os

# Set workspace and enable overwrite, make sure you update the
# workspace to your own project geodatabase.
arcpy.env.workspace = r"C:\Users\Kirkland.Aline\Documents\
ArcGIS\Projects\MyProject\MyProject.gdb"
arcpy.env.overwriteOutput = True

# Define input layers
polygon_layer = r"Facility_Polys" # Layer with facility
polygons; must have a "Category" field.
hex_grid = "hex_grid" # Hexagonal grid layer

# Define final output feature class name.
final_hex_grid = os.path.join(arcpy.env.workspace, "hex_grid_
boolean")

# Create a copy of the hex grid that we will update.
if arcpy.Exists(final_hex_grid):
    arcpy.Delete_management(final_hex_grid)
arcpy.CopyFeatures_management(hex_grid, final_hex_grid)
print("Copied hex grid to final output.")

# Get unique category values from the polygon layer.
categories = [row[0] for row in arcpy.da.SearchCursor(polygon_
layer, ["Category"]) if row[0] is not None]
categories = list(set(categories))
print("Found categories: {}".format(categories))

# Loop through each category.
for cat in categories:
    print("Processing category: {}".format(cat))

    # Create a safe field name (remove spaces/special characters;
    # you can adjust as needed).
    safe_field = cat.replace(" ", "_").replace("-", "_").
    replace("/", "_")[:20]
```

```
    # Add a new field to final_hex_grid if it doesn't already
    # exist.
    # (Here we use a SHORT integer field: 1 means at least one
    # facility is present, 0 means none.)
    field_names = [f.name for f in arcpy.ListFields(final_hex_
    grid)]
    if safe_field not in field_names:
        arcpy.AddField_management(final_hex_grid, safe_field,
        "SHORT")
        print("Added field {} to final hex grid.".format(safe_field))

    # Build the definition query to get only features for this
    # category.
    # Note: Category field is case sensitive.
    query = "Category = '{}'".format(cat)
    layer_name = "cat_layer_{}".format(safe_field)
    arcpy.MakeFeatureLayer_management(polygon_layer, layer_name,
    query)

    # Create a temporary feature class for the spatial join.
    temp_join_fc = os.path.join(arcpy.env.workspace, "temp_join_
    {}".format(safe_field))

    # Perform a spatial join.
    # Target features: final_hex_grid
    # Join features: filtered category layer (layer_name)
    # Using JOIN_ONE_TO_ONE so that each hex gets a "Join_Count"
    # field.
    arcpy.analysis.SpatialJoin(
        target_features=final_hex_grid,
        join_features=layer_name,
        out_feature_class=temp_join_fc,
        join_operation="JOIN_ONE_TO_ONE",
        join_type="KEEP_ALL",
        match_option="INTERSECT"
    )
    print("Spatial join complete for {} (output:
    {}).".format(cat, temp_join_fc))
```

```
# Build a dictionary mapping the hex grid's ObjectID
(TARGET_FID from the join) to its Join_Count.
join_dict = {}
with arcpy.da.SearchCursor(temp_join_fc, ["TARGET_FID",
"Join_Count"]) as cursor:
    for row in cursor:
        join_dict[row[0]] = row[1]

# Now update the new field in final_hex_grid.
# We set the field to 1 if the Join_Count for that hex is >
0, otherwise 0.
with arcpy.da.UpdateCursor(fnal_hex_grid, ["OBJECTID", safe_
field]) as cursor:
    for row in cursor:
        hex_id = row[0]
        if hex_id in join_dict and join_dict[hex_id] > 0:
            row[1] = 1
        else:
            row[1] = 0
        cursor.updateRow(row)

print("Updated field {} for category {}".format(safe_field,
cat))

# Clean up the temporary spatial join output and the layer.
arcpy.Delete_management(temp_join_fc)
arcpy.Delete_management(layer_name)

print("All categories processed. Final output saved as:\n{}".
format(fnal_hex_grid))
```

Step 5: You can now create a new field in your **hex_grid_boolean** layer called “Facility_Count” (the layer you output in the last step). The purpose of this new field is to have a count of the number of facility types that are accessible in each hexagon.

Figure 25: Add a facilities count field

OBJECTID	geom	geom.Length	geom.Area	Stops_NaPTAN	Retail_points	Pharmacy	Primary_schools	Play_Parks	GP	Sports_facility	Secondary_schools	Facility_Count
1	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
2	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
3	Polygon	173.20514	2165.065	1	0	0	0	0	0	0	0	1
4	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
5	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
6	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
7	Polygon	173.20504	2165.0625	1	0	0	0	0	0	0	0	1
8	Polygon	173.20504	2165.0625	1	0	0	0	0	0	0	0	1
9	Polygon	173.20514	2165.065	1	0	0	0	0	0	0	0	1
10	Polygon	173.20514	2165.065	1	0	0	0	0	0	0	0	1
11	Polygon	173.20504	2165.0625	1	0	0	0	0	0	1	0	2
12	Polygon	173.20514	2165.065	1	0	0	0	0	0	1	0	2

You can do this using the field calculator “calculate” option you see at the top of your attribute table. You then add together your existing facility fields (schools, GPs, bus stops etc.) to create your new field, e.g. !stops_NaPTAN! + !Retail_Points! etc:

Figure 26: Calculate the facility access for each hexagon grid

Calculate Field

This tool modifies the Input Table

Input Table: hex_grid_boolean

Field Name (Existing or New): Facility_Count

Field Type: Double (64-bit floating point)

Expression Type: Python

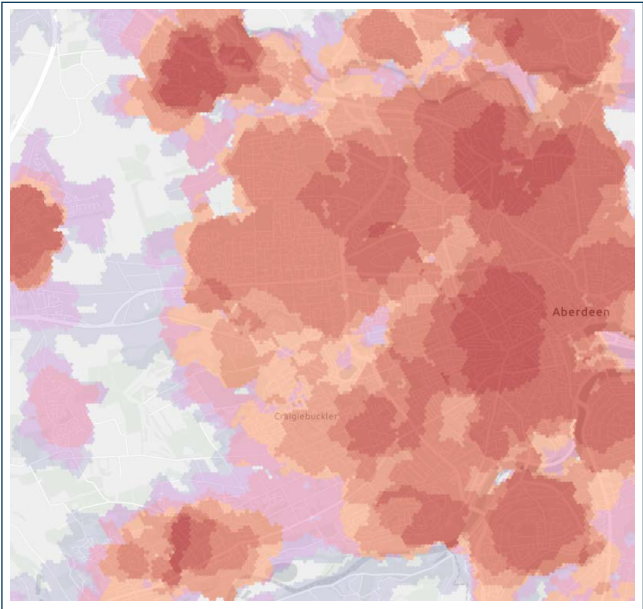
Expression: !Stops_NaPTAN! + !Retail_points! + !Pharmacy! + !Primary_schools! + !Play_Parks! + !GP! + !Sports_facility! + !Secondary_schools!

Code Block:

Enable Undo: [] Apply OK

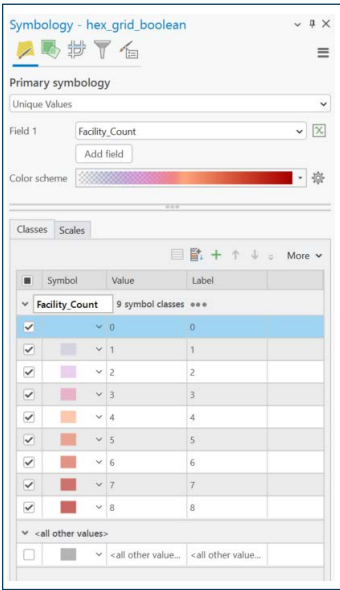
Step 6: You can now symbolise your Facility_Count field to visually see which areas have access to the greatest range of facilities:

Figure 27: Hex_grid symbolised output example



For the data visualisation, I used “Unique Values” because I wanted one colour per unique value in the Facility_Count field. I made areas with zero facilities transparent, and used a colour range from light to dark to show the concentration of different types of facilities

Figure 28: Hex_grid symbology settings



Analysis to this point may be sufficient for you. However, this analysis does not take into account whether some facilities are more important than others, or whether there are benefits to having multiple shops, cafes or other types of facilities within a 10-minute walk.

To do an analysis that takes into account these factors, you need to do the following:

After Step 3, you need to run an alternative python script that counts the number of facilities of each type that are accessible within each hexagon. The attribute table output you are going to produce will look like this:

Figure 29: Counting the number of facilities accessible in each hexagon

OBJECTID	geom	geom.Length	geom.Area	Stops_NaPTAN	Retail_points	Pharmacy	Primary_schools	Play_Parks	GP	Sports_facility	Secondary_schoo
1	Polygon	173.20504	2165.0625	29	5	2	4	4	1	14	
2	Polygon	173.20514	2165.065	25	4	2	4	4	1	11	
3	Polygon	173.20514	2165.065	26	3	2	4	4	1	14	
4	Polygon	173.20504	2165.0625	28	4	2	4	4	1	14	
5	Polygon	173.20514	2165.065	27	3	2	4	5	1	14	
6	Polygon	173.20504	2165.0625	26	4	2	4	4	1	14	
7	Polygon	173.20514	2165.065	27	4	2	4	5	1	14	
8	Polygon	173.20504	2165.0625	27	3	2	2	8	1	17	
9	Polygon	173.20504	2165.0625	29	3	2	2	8	1	17	
10	Polygon	173.20504	2165.0625	31	3	2	2	9	1	16	
11	Polygon	173.20514	2165.065	30	3	2	2	8	1	17	
12	Polygon	173.20514	2165.065	33	3	2	2	8	1	16	

You can see that from the top row of data that there are 29 bus stops, five shops, two pharmacies, four primary schools, four play parks, one GP, and 14 sports facilities within a 10-minute walk of this hexagon.

To create this output, you can use the ArcPy Script 4 .txt file, pasted into the Python window. You need to update the highlighted file path to point to your own geodatabase. You must also have a hexagonal grid layer called `hex_grid` and a polygon layer called `Facility_Polys`.

The output layer will be called **`hex_grid_combined_joined_facility_count`**.

ArcPy Script 4

```
import arcpy
import os

# Set workspace and enable overwrite. Update the workspace to
# your own project geodatabase.
arcpy.env.workspace = r"C:\Users\Kirkland.Aline\Documents\
ArcGIS\Projects\MyProject\MyProject.gdb"
arcpy.env.overwriteOutput = True

# Define input layers.
polygon_layer = r"Facility_Polys" # Layer with facility
polygons (must have a "Category" field).
hex_grid = "hex_grid" # Hexagonal grid layer.

# Define final output feature class name.
final_hex_grid = os.path.join(arcpy.env.workspace, "hex_grid_
combined_joined_facility_count")

# Create a copy of the hex grid that we will update.
if arcpy.Exists(final_hex_grid):
    arcpy.Delete_management(final_hex_grid)
arcpy.CopyFeatures_management(hex_grid, final_hex_grid)
print("Copied hex grid to final output.")

# Create a base copy of the hex grid to use for spatial joins
(so that each join uses the original geometry).
base_hex_grid = os.path.join(arcpy.env.workspace, "base_hex_
grid_for_join")
if arcpy.Exists(base_hex_grid):
    arcpy.Delete_management(base_hex_grid)
arcpy.CopyFeatures_management(hex_grid, base_hex_grid)
print("Copied base hex grid for spatial join.")

# Get unique category values from the polygon layer.
categories = [row[0] for row in arcpy.da.SearchCursor(polygon_
layer, ["Category"]) if row[0] is not None]
categories = list(set(categories))
print("Found categories: {}".format(categories))
```

```

# Loop through each category.
for cat in categories:
    print("Processing category: {}".format(cat))

    # Create a safe field name (remove spaces/special characters;
    # adjust as needed).
    safe_field = cat.replace(" ", "_").replace("-", "_").
    replace("/", "_")[:20]

    # Add a new field to final_hex_grid if it doesn't already
    # exist.
    field_names = [f.name for f in arcpy.ListFields(final_hex_
    grid)]
    if safe_field not in field_names:
        arcpy.AddField_management(final_hex_grid, safe_field,
        "SHORT")
        print("Added field {} to final hex grid.".format(safe_field))

    # Build the definition query to select only features for this
    # category.
    query = "Category = '{}'".format(cat)
    layer_name = "cat_layer_{}".format(safe_field)
    arcpy.MakeFeatureLayer_management(polygon_layer, layer_name,
    query)

    # Create a temporary feature class for the spatial join.
    temp_join_fc = os.path.join(arcpy.env.workspace, "temp_join_
    {}".format(safe_field))

    # Perform a spatial join using the base_hex_grid as the
    # target.
    arcpy.analysis.SpatialJoin(
        target_features=base_hex_grid,
        join_features=layer_name,
        out_feature_class=temp_join_fc,
        join_operation="JOIN_ONE_TO_ONE",
        join_type="KEEP_ALL",
        match_option="INTERSECT"
    )
    print("Spatial join complete for {} (output:
    {})".format(cat, temp_join_fc))

```

```

    # Build a dictionary mapping the hex grid's ObjectID
    # (TARGET_FID from the join) to its Join_Count.
    join_dict = {}
    with arcpy.da.SearchCursor(temp_join_fc, ["TARGET_FID",
    "Join_Count"]) as cursor:
        for row in cursor:
            join_dict[row[0]] = row[1]

    # Update the new field in final_hex_grid with the actual
    # count.
    with arcpy.da.UpdateCursor(final_hex_grid, ["OBJECTID", safe_
    field]) as cursor:
        for row in cursor:
            hex_id = row[0]
            row[1] = join_dict.get(hex_id, 0) # Use the join
            count (or 0 if not found).
            cursor.updateRow(row)

    print("Updated field {} for category {}".format(safe_field,
    cat))

    # Clean up the temporary spatial join output and the layer.
    arcpy.Delete_management(temp_join_fc)
    arcpy.Delete_management(layer_name)

print("All categories processed. Final output saved as:\n{}".
    format(final_hex_grid))

```

With a count of different type of facilities accessible from each hexagon, you are likely to have some grids with a higher count of facilities than is likely to be meaningful when considering a 20-minute neighbourhood. For example, in central Aberdeen, some hexagons will have access to more than 100 bus stops within a 10-minute walk, many of these stops are likely for the same buses but are just somewhat further down the street. You can handle this overrepresentation of certain facilities by **weighting your data**.

Weighting your facilities also gives you the opportunity to assign a relative importance to different facilities. For example, people may visit a shop most days, and consider it essential to their daily life. You can give this facility a higher number than a café when assigning weights so that neighbourhoods that only have a shop, but no café would score higher than a neighbourhood with only a café and no shop. Suggested weightings are provided on page 8.

You can use the field calculator to add new weighted columns to your data. An example for bus stops is included below:

- If a hexagon only has access to one bus stop, the bus stop field is given a score of 0.5.
- If a hexagon has access to two or more bus stops, then the bus stop field is given a score of 1.
- If a hexagon has no access to a bus stop, it is given a score of 0.

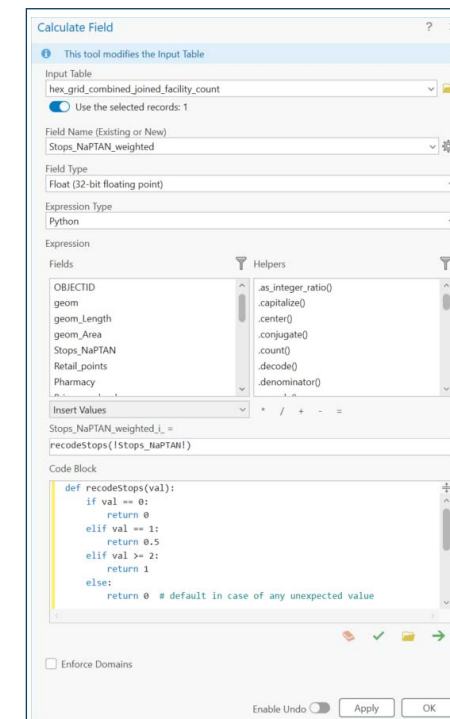
The relevant field calculator code is included here:

```
recodeStops(!Stops_NaPTAN!)

def recodeStops(val):
    if val == 0:
        return 0
    elif val == 1:
        return 0.5
    elif val >= 2:
        return 1
    else:
        return 0 # default in case of any unexpected value
```

Note: when using the field calculator for calculations, make sure you use an appropriate field type, for example a Float (32-bit floating point) if you will have decimal places in your values.

Figure 30: Calculate weighted field values



You can use this code structure to add weightings to all your facility types, to reflect their relative importance.

Once this is complete, you can calculate a weighted facilities field by adding together all of your weighted facility fields to create a final score for each hexagonal grid in your dataset, e.g. !Retail_points_weighted! + !Pharmacy_weighted! etc.:

Figure 31: Calculate weighted facility field

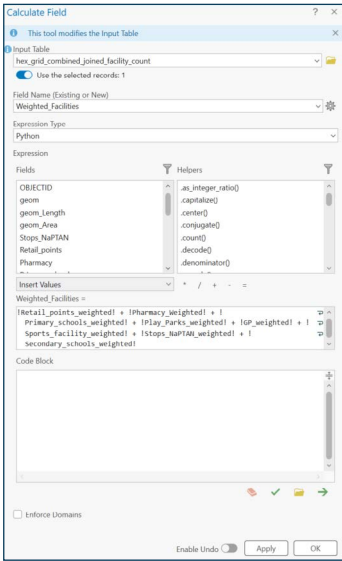
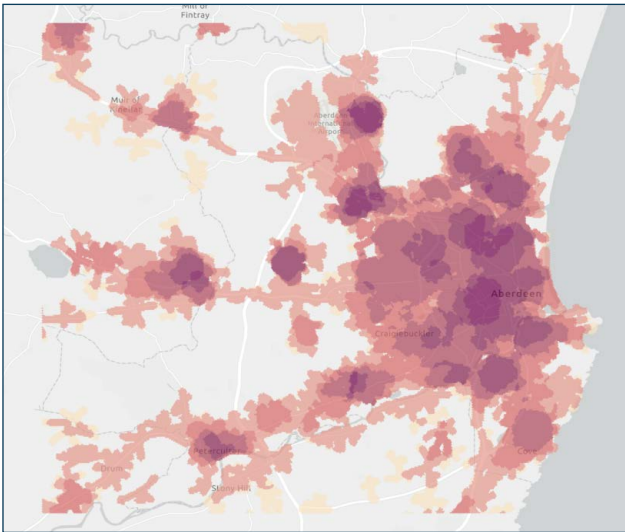


Figure 32: Output attribute table showing final weighted facility field

hex_grid_combined_jo..._facility_count							
	Primary_schools_weighted	Play_Parks_weighted	GP_weighted	Sports_facility_weighted	Stops_NAPTAN_weighted	Secondary_schools_weighted	Weighted_Facilities
1	1	1	1	1	1	0.8	7.8
2	1	1	1	1	1	0.8	7.8
3	1	1	1	1	1	0.8	7.8
4	1	1	1	1	1	0.8	7.8
5	1	1	1	1	1	0.8	7.8
6	1	1	1	1	1	0.8	7.8
7	1	1	1	1	1	0.8	7.8
8	1	1	1	1	1	0.8	7.8
9	1	1	1	1	1	0.8	7.8
10	1	1	1	1	1	0.8	7.8
11	1	1	1	1	1	0.8	7.8
12	1	1	1	1	1	0.8	7.8

Finally, you can visualise the data on your map:

Figure 33



This layer can now be added to an ArcGIS Online webmap so it can be shared across your local authority or with members of the public.

OPTIONAL FURTHER ANALYSIS

You can enhance this analysis by adding additional layers of information so you can understand the impact of accessibility on different land uses. For example:

- Adding housing land supply data to monitor the accessibility of housing sites.
- Adding employment land supply data to review potential access benefits if certain sites are developed.
- Adding Scottish Index of Multiple Deprivation (SIMD) data, to compare accessibility in areas with higher and lower SIMD scores.
- Adding population density information to identify areas with high population density, but poor access to services.
- Adding street lighting columns to assess which areas may be poorly lit at night as this can impact on sense of safety.

Consultation and engagement: Engaging with local communities to understand more about their perceptions of accessibility is recommended as a way to enhance and build on the findings of your analysis. Qualitative engagement is recommended to understand whether there are concerns or challenges not captured within the analysis (for example, people may feel unsafe on certain routes). You may wish to do this as part of your local development plan engagement process.



APPENDIX

Theme	Facility type	Highland Council	West Dunbartonshire Council
Education	Nursery	X	X
Education	Primary School	X	X
Education	Secondary School	X	X
Education	Further Learning		X
Shopping	Supermarket	X	X
Shopping	Local shop	X	
Shopping	Key retail areas		X
Health	GP	X	X
Health	Hospital	X	X
Health	Pharmacy	X	X
Health	Care Homes		X
Transport	Bus stop	X	X
Transport	Train station	X	X
Transport	Ferry	X	
Play	Children's play area	X	X
Environment	Woodland	X	
Environment	Park		X
Postal Services	Post Office	X	
Financial Services	ATM	X	
Postal Services	Post Locker	X	
Community Facilities	Community Centre	X	X
Sports Facilities	Sports Facility	X	
Sport Facilities	Leisure Centre	X	X
Community Facilities	Library	X	X
Religious Facilities	Church	X	
Religious Facilities	Mosque	X	

Theme	Facility type	Highland Council	West Dunbartonshire Council
Religious Facilities	Synagogue	X	
Religious Facilities	Hindu Temple	X	
Waste Management	Recycling/Waste Facility	X	
Financial Services	Bank/Building Society	X	
Food and Drink	Pub	X	
Food and Drink	Restaurant	X	
Food and Drink	Café	X	
Food and Drink	Takeaway	X	
Community Facilities	Allotment	X	X
Cultural Facilities	Museum	X	
Cultural Facilities	Theatre	X	
Cultural Facilities	Concert Hall	X	
Shopping	Other shops	X	
Sport Facilities	Golf course		X
Sport Facilities	Tennis court		X
Sport Facilities	Bowling green		X
Sport Facilities	Bathing beach		
Sport Facilities	Swimming pool		X
Sport Facilities	Football/rugby/shinty pitch		X
Entertainment	Cinema		

For help and support with the analysis described in this document, please reach out to the IS Spatial Hub:
spatialhub@improvementservice.org.uk

Improvement Service
West Lothian Civic Centre
Howden South Road
Livingston
EH54 6FF

T: 01506 282012
E: info@improvementservice.org.uk
W: www.improvementservice.org.uk