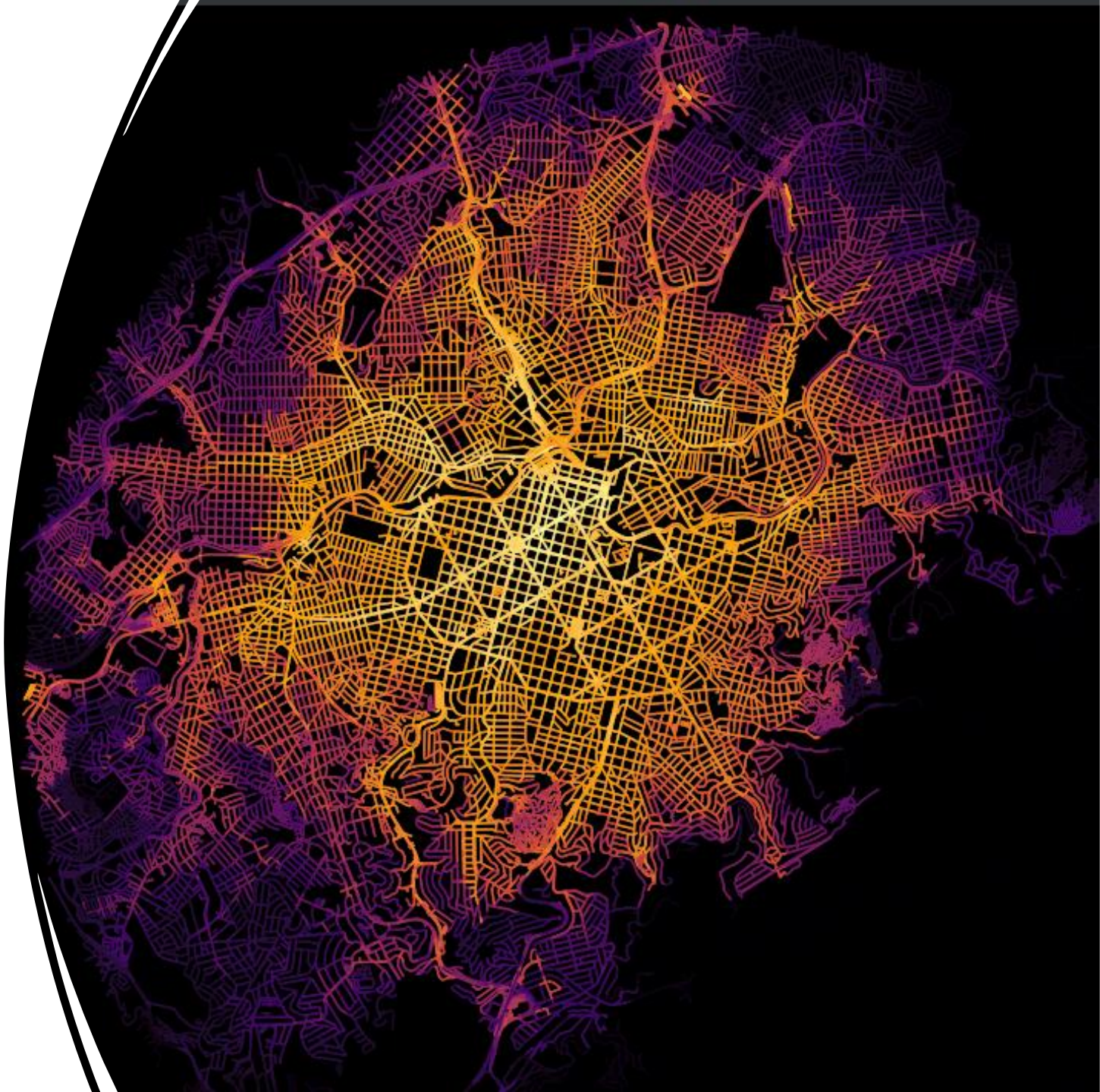
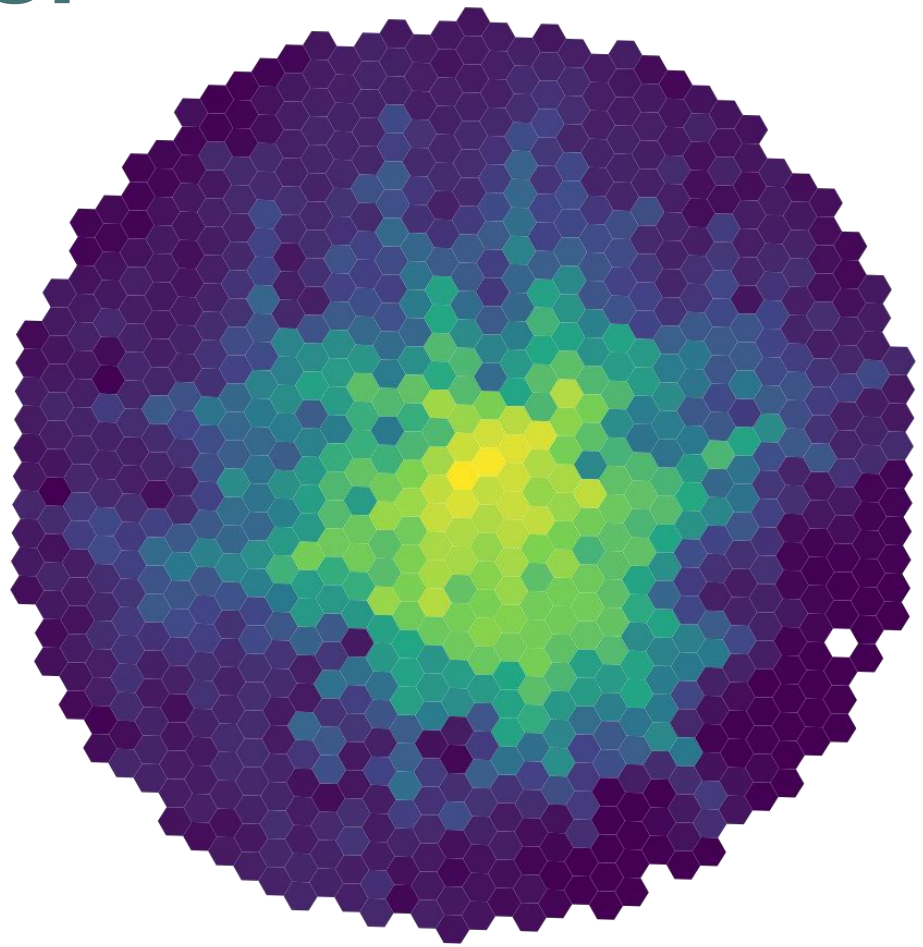
Computational requirements



Computational requirements:

1. A computer
   - No super-computer
   - 8 GM RAM is enough for most cases



2. A routing engine / algorithm

   Good news!

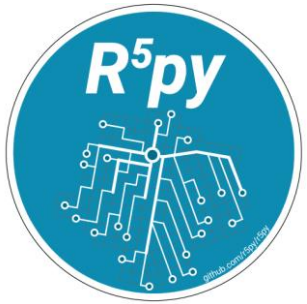
   Growing number of open-source tools

ipea

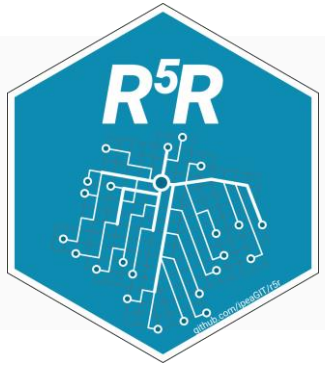
Routing engines

* network analysis packages

Our tool of choice:



ipea



r5r: rapid realistic routing with R5 in R

CRAN 2.0 | downloads 912K | DOI 10.32866/001c.21262 | GitHub code

https://ipeagit.github.io/r5r

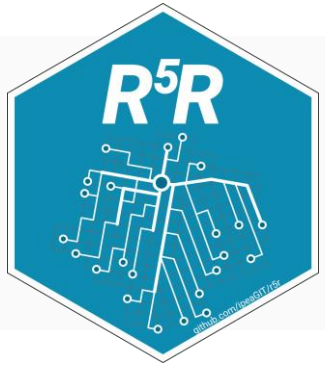
R + R5

Conveyal

Fast routing on multimodal transport networks

- Travel time / distance matrices
- Accessibility
- Trip planning
- Isochrones
- Level of traffic stress for cycling
- Terrain elevation
- Monetary costs
- Time window

ipea



Blazing fast

Table 3. Performance Analysis Results

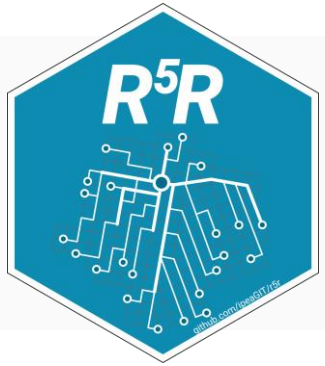
Activity	Scenario	ArcGIS Pro	R-OTP	Python-OTP	R5R	Emme
Make/Build Network	All	11m 26s	8m 19s	7m 24s	2m 39s	1m 25s
Calculate OD Matrix and Accessibility	1 x 100	1m 37s	20s	23s	<1s	2m 23s
	1 x 1,000	1m 47s	2m 49s	24s	<1s	4m 46s
	1 x 10,000	2m 51s	22m 37s	28s	1s	28m 36s
	1 x 100,000	12m 59s	7h 12m 15s	1m 4s	7s	*not run*
Calculate OD Matrix and Accessibility	100 x 100	3m 24s	25m 30s	2m 2s	1s	5m 5s
	1,000 x 1,000	6m 56s	>24h (DNF)	13m 42s	9s	18m 33s
	10,000 x 10,000	49m 53s	*not run*	4h 49m 18s	7m 50s	13h+
	100,000 x 100,000	*not run*	*not run*	*not run*	18h 15m 43s	*not run*

r5r is *x* times faster than:

Number of OD pairs	ArcGis	OTP-Py	Emme
100^2	204	122	305
1.000^2	46	91	124
10.000^2	6	37	100
100.000^2	*Inf*	*Inf*	*Inf*

Higgins et al (*2022*) Calculating Place-Based Transit Accessibility: Methods, Tools and Algorithmic Dependence. Journal of Transport and Land Use https://jtlu.org/index.php/jtlu/article/view/2012

ipea



r5r: rapid realistic routing with R5 in R

CRAN 2.0 | downloads 912K | DOI 10.32866/001c.21262 | GitHub code

https://ipeagit.github.io/r5r

New version **v2.0**

It requires **Java JDK 21** : several open-source providers – Link

Check your Java version with:

```
rJava::.jinit()
rJava::.jcall("java.lang.System", "S", "getProperty", "java.version")
```

```
[1] "21.0.1"
```

If your output points to a JDK 11, you can install the older version of {r5r}

```
devtools::install_version(package = 'r5r',
                          version = '1.1.0')
```


ipea

A crash course on

Urban accessibility with R

Rafael H. M. Pereira

@UrbanDemog

ipea Institute for Applied Economic Research