# How to GIS in Python

A tale of two Cities

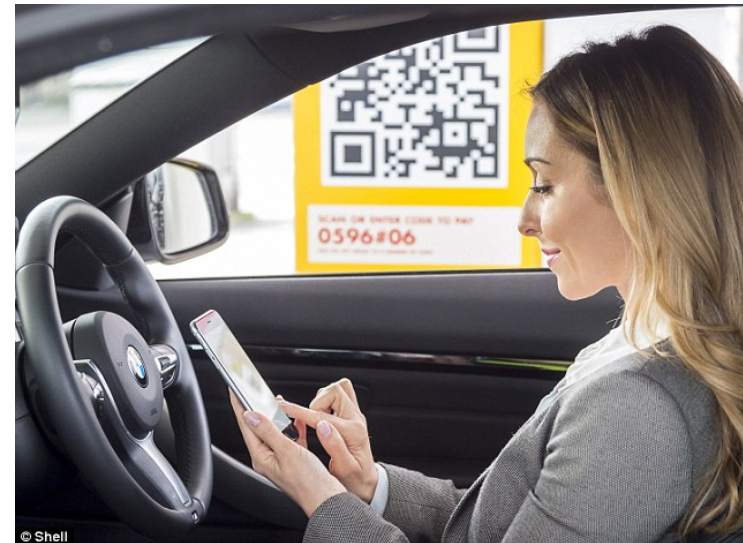Anders Lehmann, Aarhus University

# Who am I?

- M.Sc.EE 1989 from Technical University of Denmark
- 15 years R&D at Bang & Olufsen
- PyPy – EU Research project under FP6
  - Saarbrücken, Germany
- Startup company – AI
- SW-Consultant – Vestas
- Lecturer at Aarhus University

# EcoSense

- Collective Mobile Sensing

- Modeling of Emissions

  – Climate gasses

  – Pollution

- Visualization

# Contents

- What is GIS
- Applications of GIS
- Data sources for GIS
- Python tools
- Python examples
- Istanbul transport model
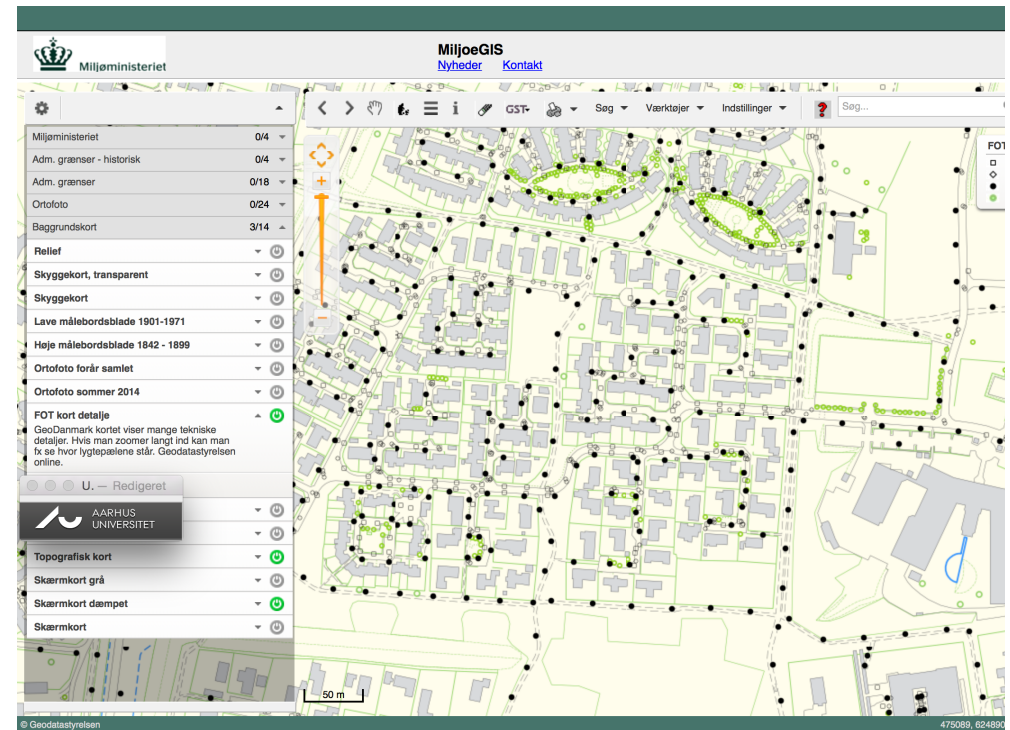
# GIS Introduction

- GIS is about maps – Simple!
- But how to make 2 dimensional maps from a sphere?
- Lots of ways!
- Which can make GIS complex

# Projections

- About 4000 different projection
- Mostly a solved problem
- Just use WSG84

# GIS Applications

- Keep track of assets
  - Underground cabling
  - Sewage
  - Fleet management
- Zoning
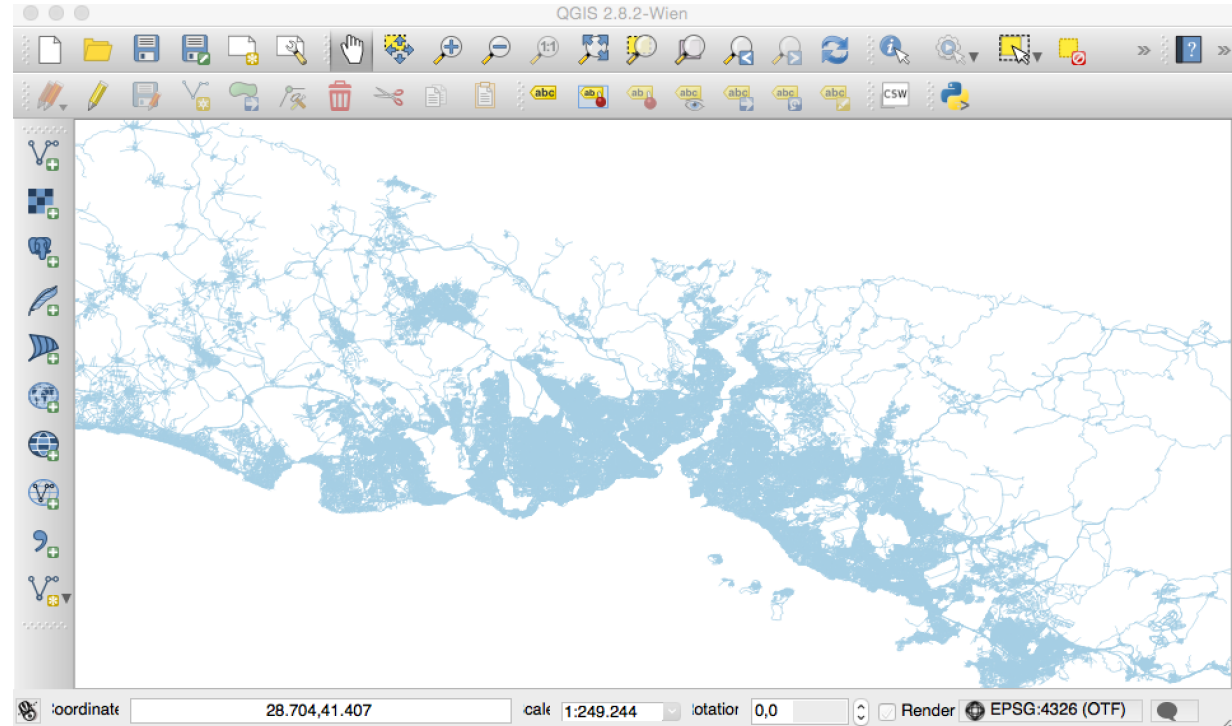- Planning

# GIS Applications

- Routing (Google Maps, Tom Tom, Yandex,…)
- City planning
- Traffic modeling

# GIS Data

- Open Street Map
- Google Earth
- National data centers
- Local data resources
  - Municipalities

# Python tools

- QGIS
- ArcGIS

# GIS tools

- Postgres/PostGIS/pgrouting
- GDAL
- OGR2OGR

# Examples

- Open Street Map

Section 2

# TRAFFIC MODELING

# Traffic modeling

Predict traffic flows

Model how drivers choose routes

Assignment problem

Derived from econometrics

Looking for Equilibrium/Steady state solutions
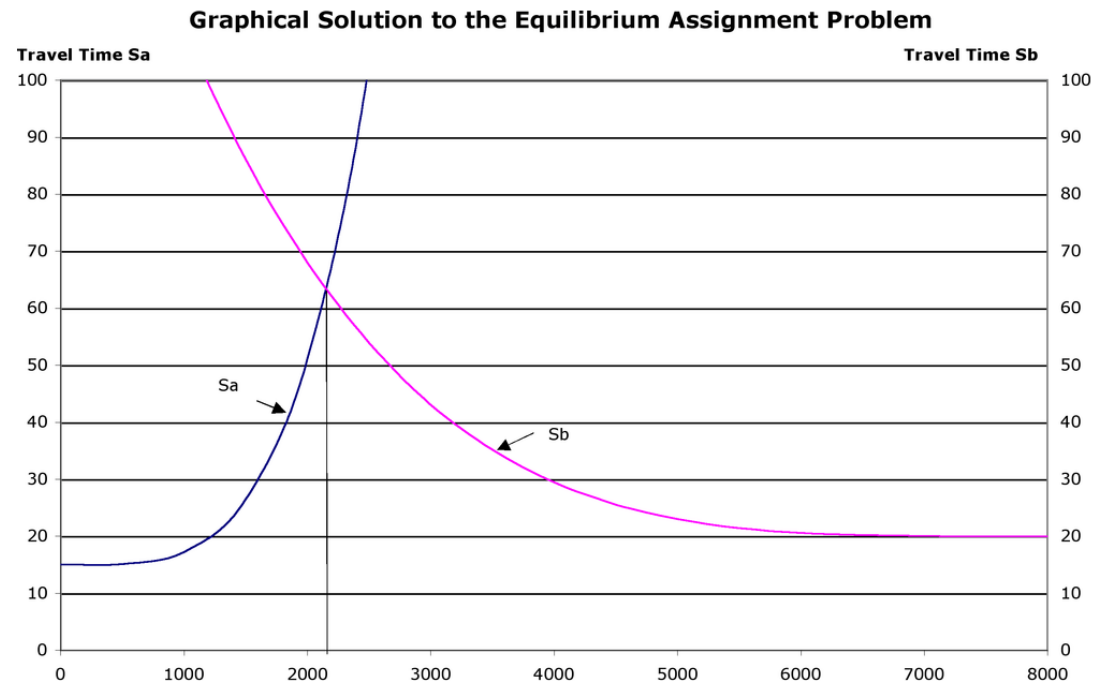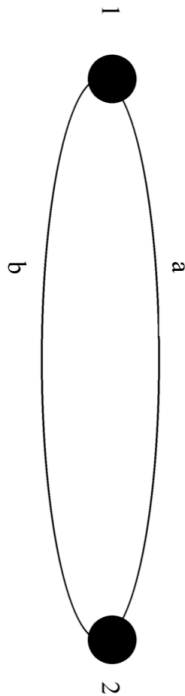
# Traffic modeling

# Traffic model

- Congestion model
  - Travel time dependent on congestion
  - BPR formula (Bureau of Public Roads)

$$S_a\left(v_a\right) = t_a\left(1 + 0.15\left(\frac{v_a}{c_a}\right)^4\right)$$

# Traffic model

- ## User Equilibrium
  - – All drivers have the shortest travel time
  - – All must have the same travel time



### Graphical Solution to the Equilibrium Assignment Problem

# Traffic model

- Stochastic User Equilibrium
  - All drivers THINK they have the shortest travel time
  - Still use the BPR, but add a stochastic error term
  - Leads to GEV formulation of the assignment problem
  - Ie. Path Size Logit

$$P_k = \frac{\exp\left(V_k + \beta_{PS} \cdot \ln PS_k\right)}{\sum_{l \in C} \exp\left(V_l + \beta_{PS} \cdot \ln PS_l\right)}$$

# Istanbul case

- Data from Open Street Map
- Converted to form a topology
- 300.000 road segments (bidirectional)
- 2000 Origin Destination pairs
- Postgres/PostGIS/pgRouting
- Python for driving the Queries
- QGIS for visualization

# Demo

# Conclusion

- Python can definitely be used in GIS
- There are many tools (a jungle?) available
- There are many open data sources
- There is a learning curve to understand the terminology

# Thank You!

- @redorlik
- anders@hih.au.dk