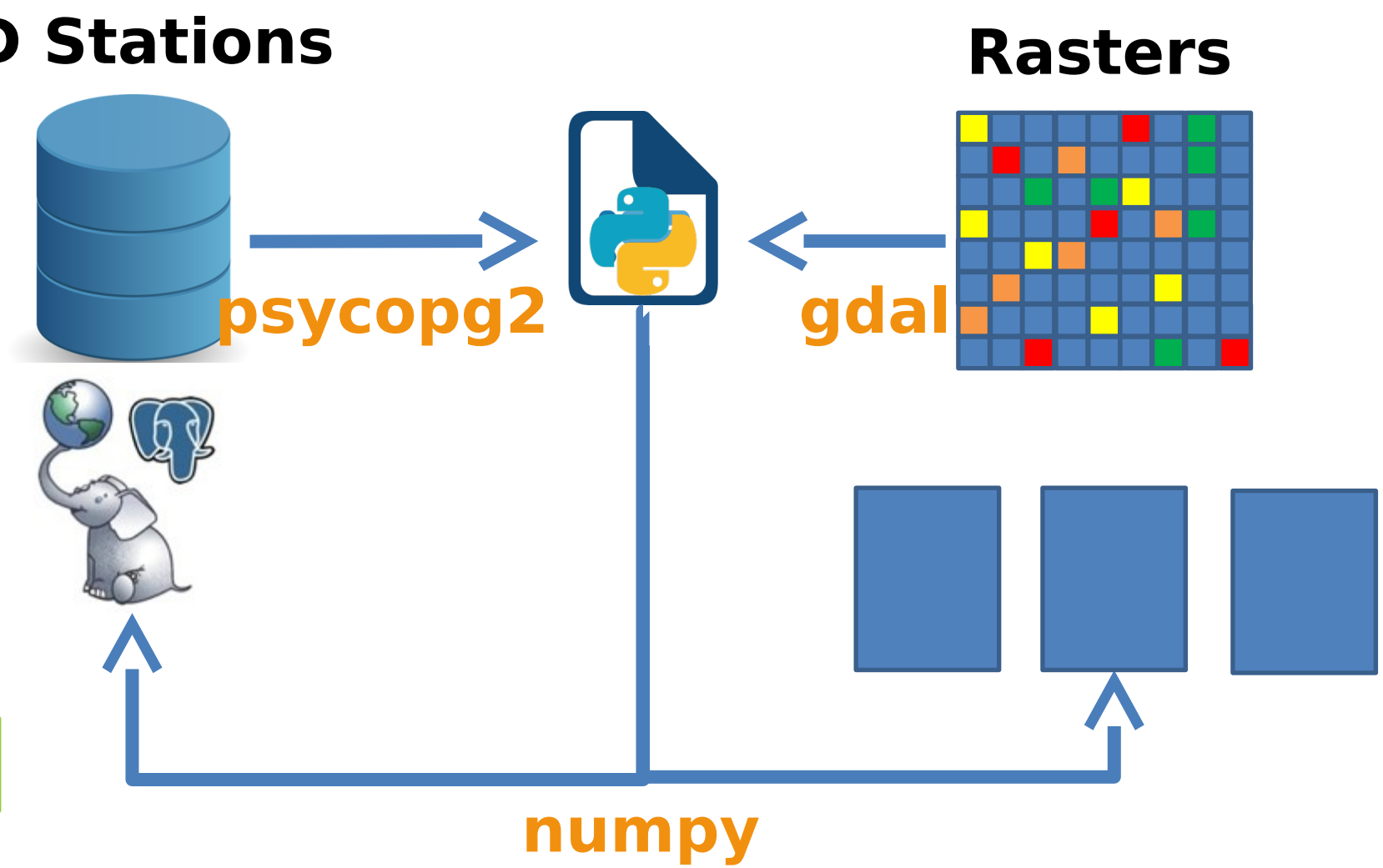
# *From ski area spatial modelling to snowpack viability*

**H. François**
**P. Spandre, S. Morin et E. George**
**PySciDataGre, Grenoble, 8 février 2019**

UMR 3589

# Framework



# Basics

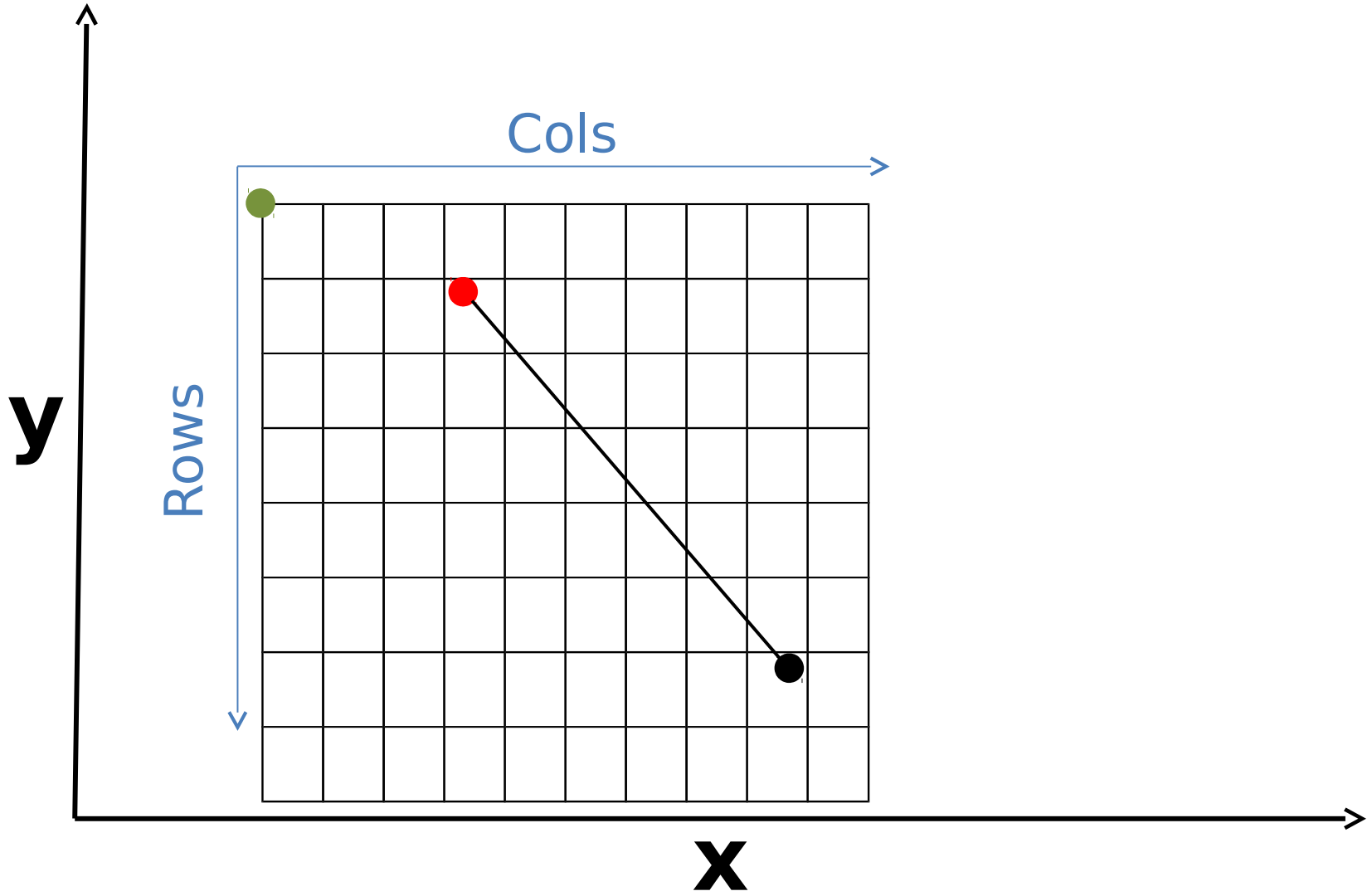
- Open rasters
  - dst_ds = gdal.Open(raster)
  - gt = dst_ds.GetGeotransform()
  - band = dst_ds.GetRasterBand(1)
  - array = band.ReadAsArray()

- Filter and get vectorial data using SQL / spatial SQL

METEO FRANCE

irstea

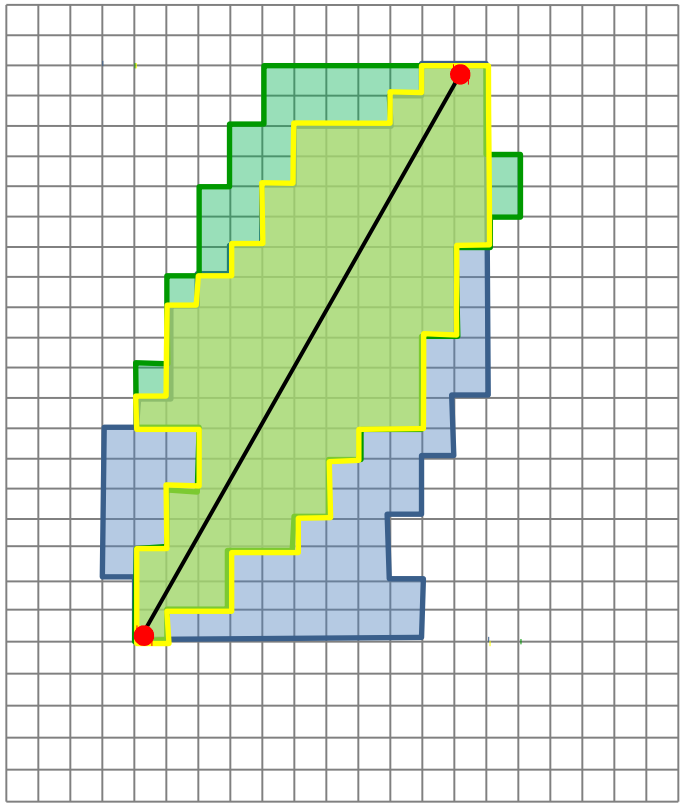
# How do we locate in space ?



# - 1 -

# Ski area spatial modelling

# From ski area to gravity area



**All points reachable from the top of a ski-lift allowing skier to go back to the bottom of any ski lift within the same ski resort**

- Spatial  information crossing
- Skier inertia when he goes down
- « non skiable » areas are exclude (slopes)
- Agregate at the scale of a resort

# The use of recursivity to select pixel in a cascade

```
def detect_down:
ref_mat = height[rmin:rmax,cmin:cmax]
tmptarget = target[rmin:rmax,cmin:cmax]
tmpsize = tmptarget.shape
targetzero = numpy.zeros((tmpsize[0], tmpsize[1]),
numpy.int8)
targetzero = targetzero + 1 *
numpy.logical_and(numpy.logical_and(numpy.logical_and(
(ref_mat - height[ref_p] < 0),tmptarget == 0),
ref_slopes <= slopemax), ref_mat >= min

for n in numpy.argwhere(targetzero==1):
    p = (n[0] + rmin, n[1] + cmin)
    target[p] = 1
    new_pt.append(p)
return new_pt
def ski_slope_down:
    np = detect_down
    for p in np:
        ski_slope_down
```

# Polygonization: create raster and shapefile

```
#create tmp raster
driver = gdal.GetDriverByName("GTiff")
skirast = driver.Create('test.tif', h, w, 1,
gdal.GDT_Byte)
skirast.SetGeoTransform(mnt_trans)
srs = osr.SpatialReference()
srs.ImportFromEPSG(2154)
skirast.SetProjection(srs.ExportToWkt())
skirast.GetRasterBand(1).WriteArray(fromtop)

#create tmp shapefile for poligonyzation
driver = ogr.GetDriverByName('ESRI Shapefile')
tmpshp = driver.CreateDataSource('tmp.shp')
tmplayer = tmpshp.CreateLayer('tmp', srs)
tmplayer.CreateField(ogr.FieldDefn("val",
ogr.OFTInteger))

gdal.Polygonize(skirast.GetRasterBand(1), None,
tmplayer, 0, [], None)
```

METEO FRANCE

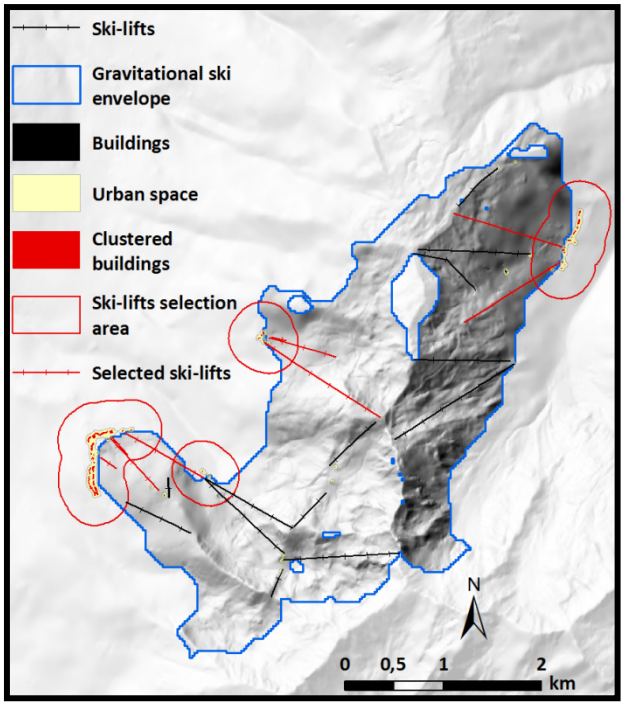
irstea

# Polygonisation: insert into BD Stations

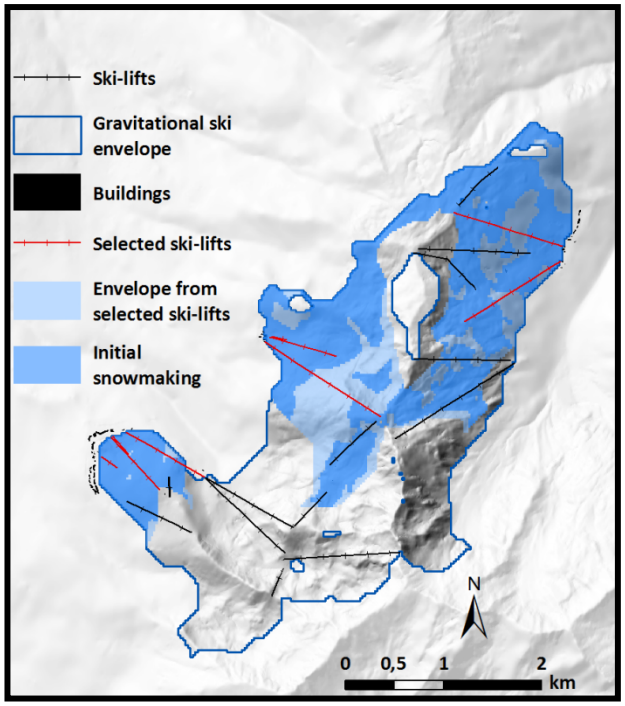
```
tmplayer.SetAttributeFilter("val = 1")
j = 0
for feat in tmplayer:
    geom = feat.GetGeometryRef()
    if j == 0 and geom.GetArea() > 3125:
        geom_wkb = geom.ExportToWkb()
        geom = ogr.CreateGeometryFromWkb(geom_wkb)
        fromtop_final = geom
        j = 1
    elif j == 1 and geom.GetArea() > 3125:
        geom_wkb = geom.ExportToWkb()
        geom = ogr.CreateGeometryFromWkb(geom_wkb)
        fromtop_final = fromtop_final.Union(geom)
fromtop_wkt = fromtop_final.ExportToWkt()
query = """insert into
stations.geo_dsa_building(ind,id_rm,fromtop_geom)
select %s::varchar, %s::varchar,
ST_GeomFromText(%s,2154)"""
cur.execute(query,(sta[0],id_rm,fromtop_wkt,))
myconn.commit()
```

# Spatial modelling of snowmaking area

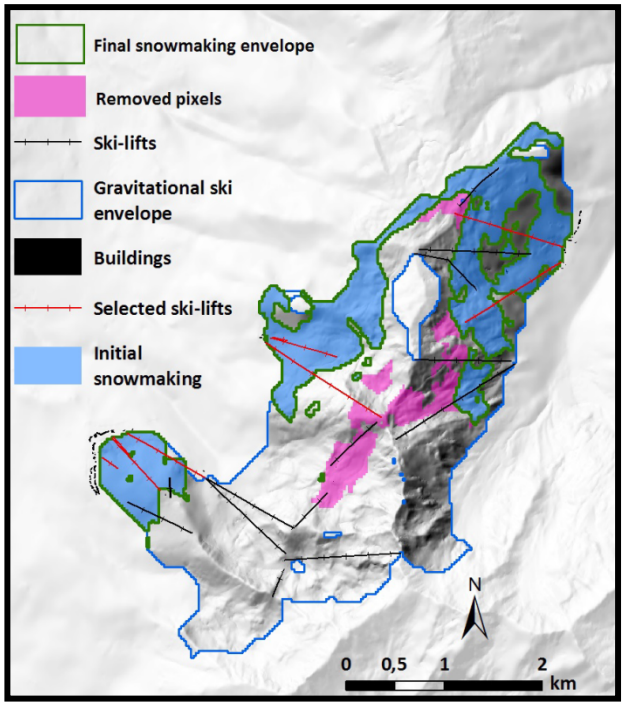
1. Snowmaking near leisure real estate




2. Gravity area and slopes



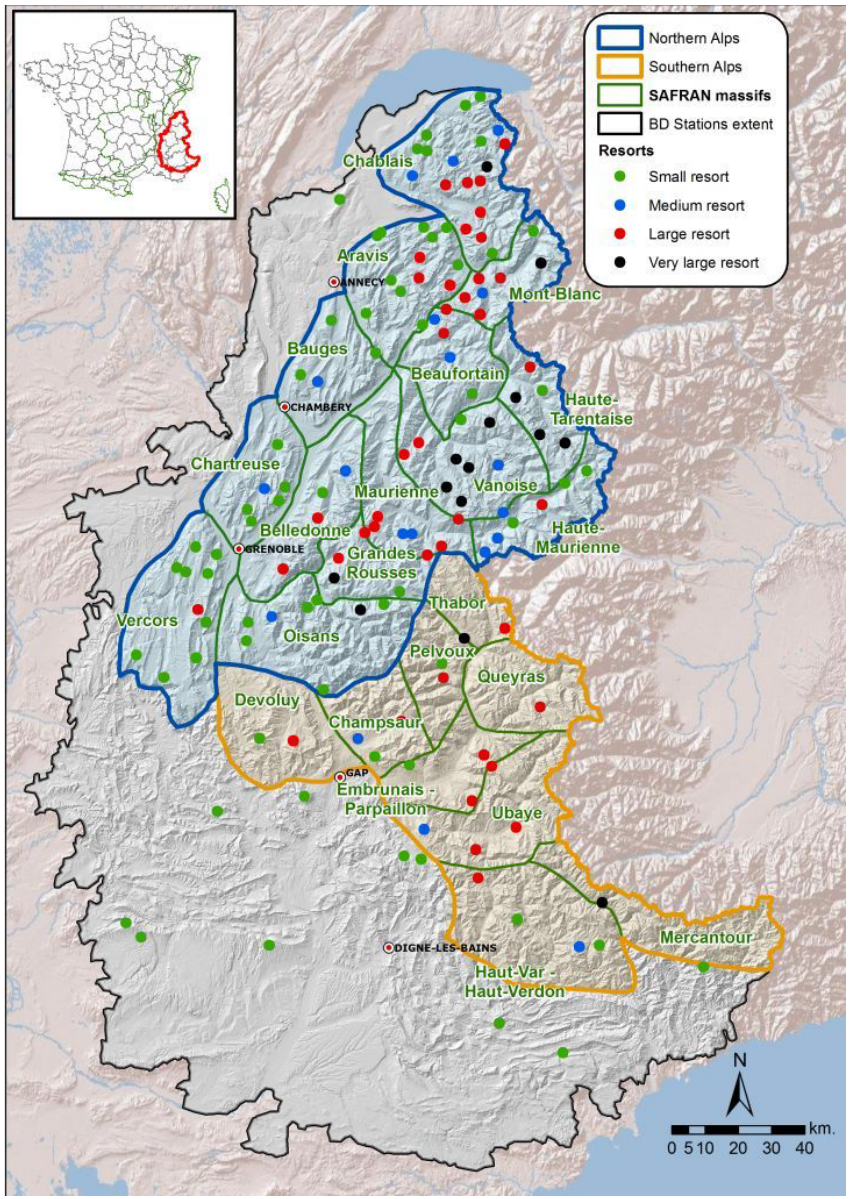

3. Adjust to target



# - 2 -

# Cross with snow data

# Basics




## 1. SAFRAN Mountain massifs

- 23 massifs around 900 km²
- Elevation with 300m step

## 2. BD Stations

- 148 alpine ski resorts
- Spatial information

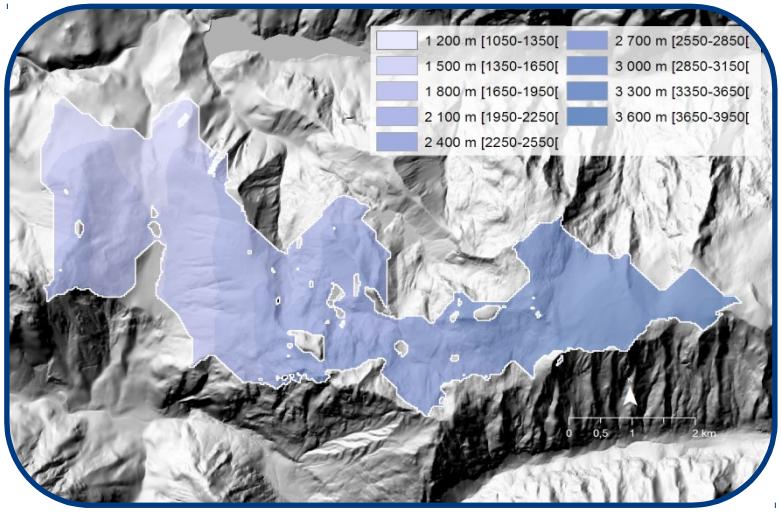
# Spatial interpolation of snow simulations

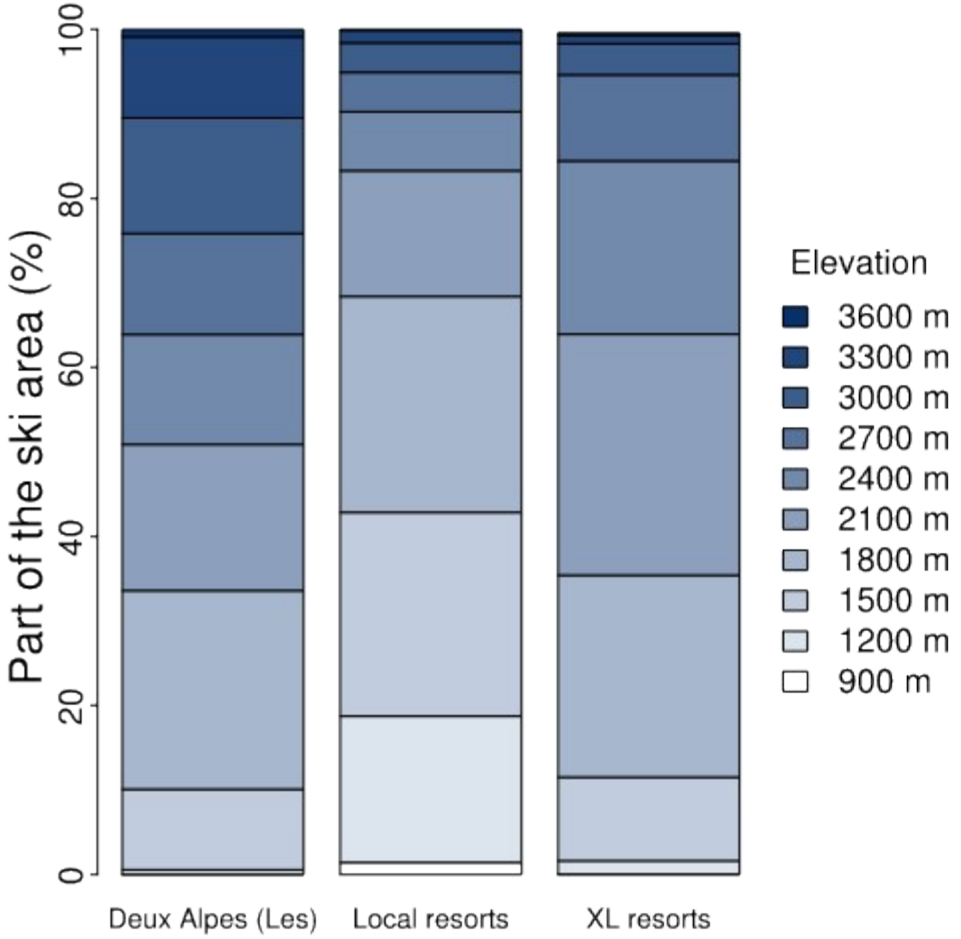
- Reclass DEM and its derivatives
- Identify localization type regarding standard values of mountain range, elevation, slope and aspect
- Query the BD Stations database to retrieve snow data at different dates
- Data storage into a numpy array
- Output
  - Viability index
  - Snow cover raster

METEO FRANCE

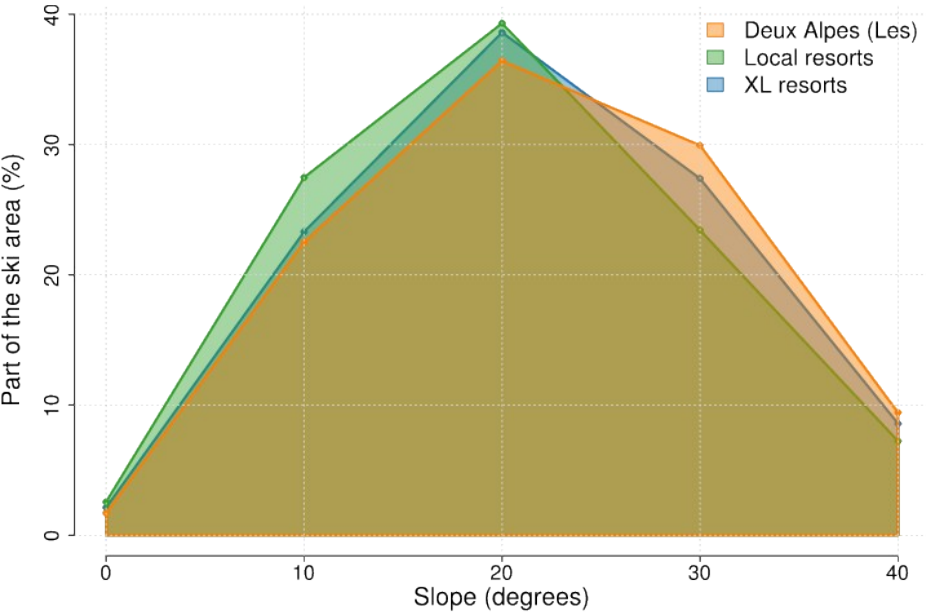
irstea

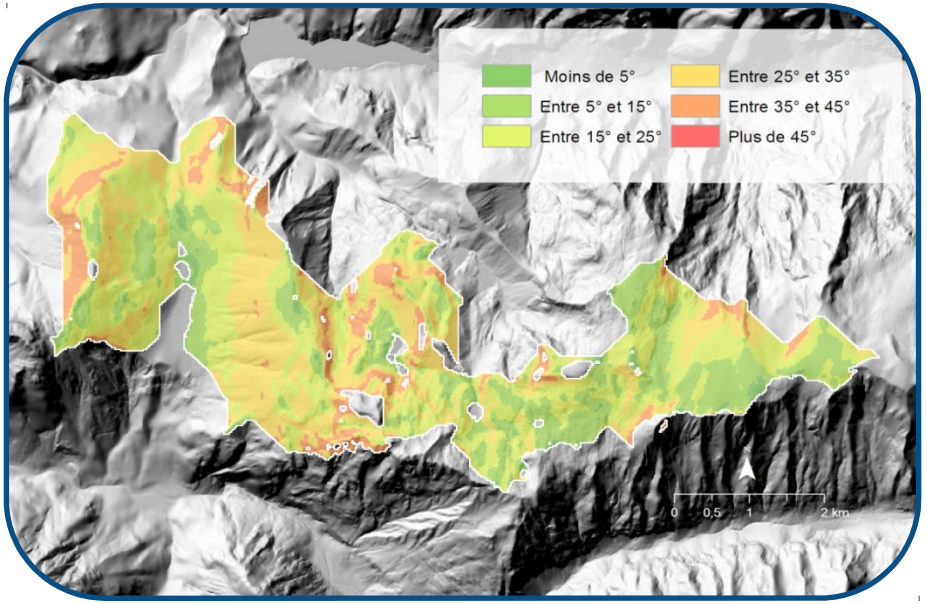
# Ski area properties: elevations



# Ski area properties: slopes

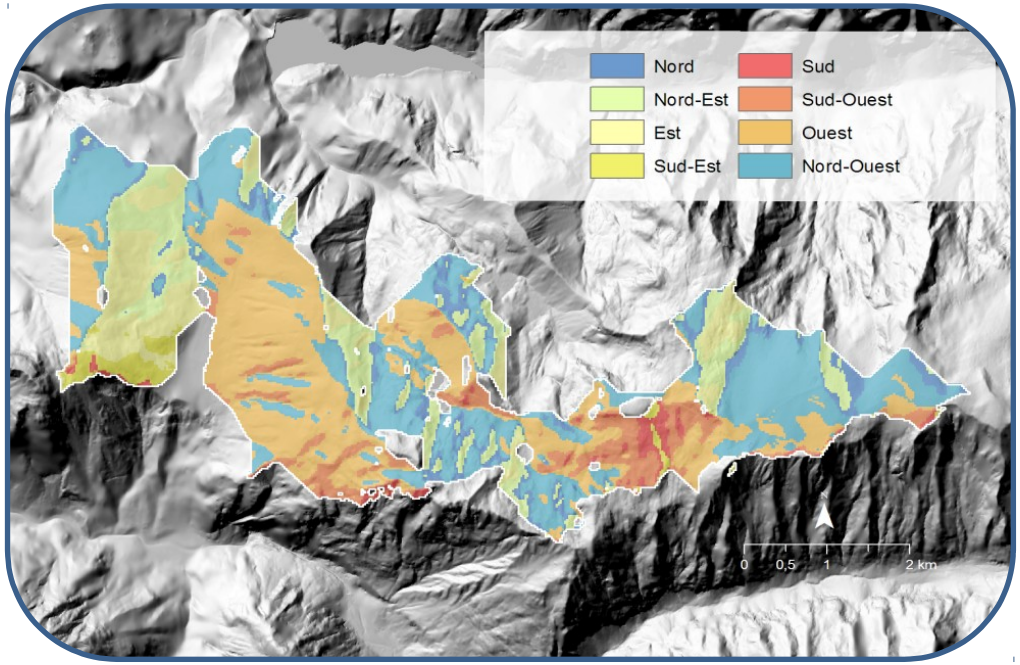


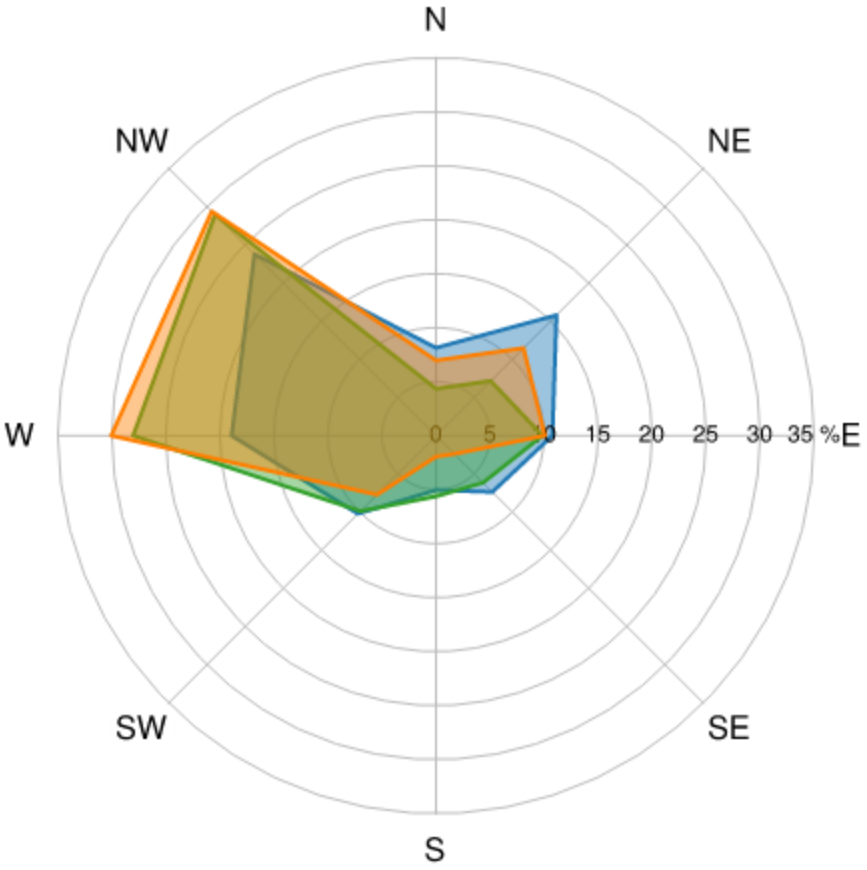




METEO FRANCE

irstea

# Ski area properties: elevations




METEO FRANCE

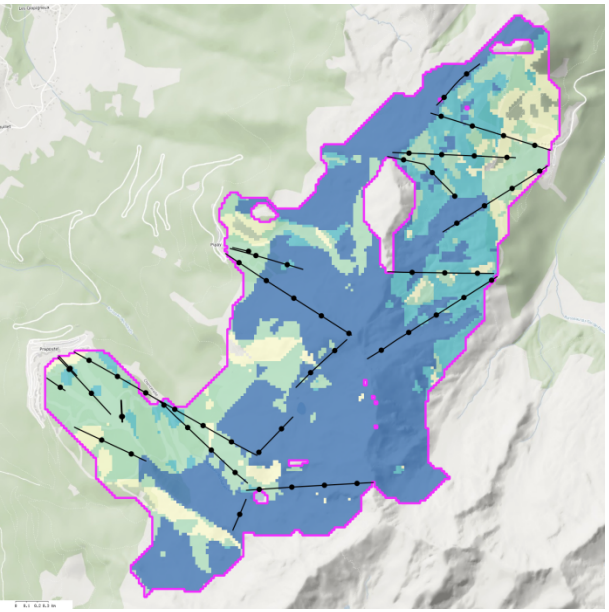
irstea

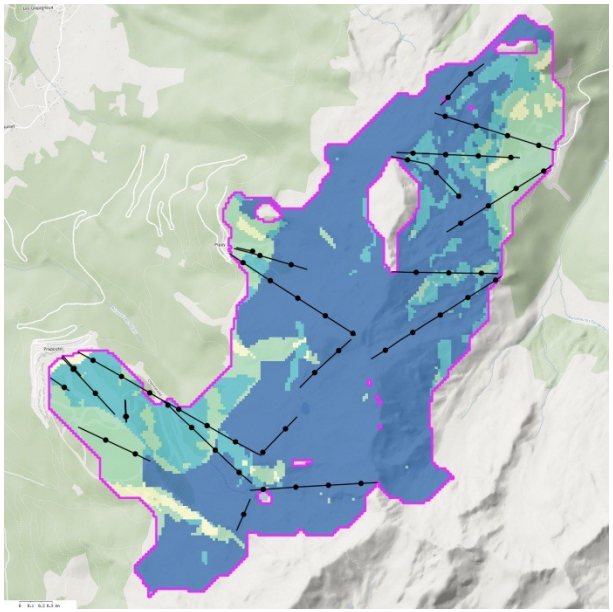
# Snow cover in 2006-2007

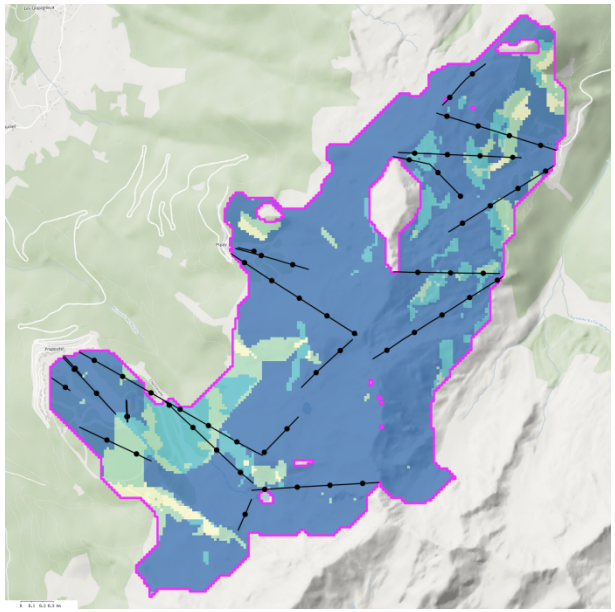
1. Natural snow

2. Groomed snow

3. Managed snow







METEO FRANCE

irstea

< 50 days

< 80 days

< 100 days

> 100 days

Gravitational Ski Envelope

Ski Lifts