



Travail de session : Statistiques spatiales

1 Avril 2020

Jean-Philippe Boutin - 9532697

Paul Conerardy - 11206073

80-619: Méthodes avancées en exploitation de données
Maîtrise en gestion (M. Sc.) Intelligence d'affaires
Professeur(e)s : Aurélie Labbé et Denis Larocque

Contents

1 Présentation du sujet	2
2 Revue de littérature	2
2.1 Introduction/sources générales	2
2.2 Branches majeures des statistiques spatiales	3
2.2.1 Type de données ponctuelles (spatial point patterns)	3
2.2.2 Type de données latticielles (lattice and areal unit data)	3
2.2.3 Type de données géostatistiques (continuous spatial variation data)	3
2.3 Des références pour quelques domaines d'application	4
2.3.1 Écologie	4
2.3.2 Géosciences	4
2.3.3 Épidémiologie	4
3 Méthodes	4
3.1 Modèle spatial général	4
3.2 Données ponctuelles (point pattern analysis)	5
3.3 Les données latticielles (discrete spatial variation, including lattice and areal unit data)	5
3.4 Les données géostatistiques (geostatistical - continuous spatial variation)	6
4 Ressources R	8
4.1 Introduction	8
4.2 Catégories principales des librairies R en statistiques spatiales	8
4.3 Les 30 librairies en statistiques spatiales les plus téléchargées	10
4.4 Quelques références R incontournables en analyse spatiale	12
4.5 Librairies abordées durant la section tutoriel	12
5 Exemples d'analyses avec des données	14
5.1 Processus de points	14
5.1.1 Librairie Spatstat	14
5.1.2 Covariables	15
5.1.3 Intensité	15
5.1.4 Planar Point Pattern	17
5.1.5 Processus de points de Poisson	17
5.1.6 Processus groupés et ordinaires	18
5.1.7 Distribution des plus-proches voisins	19
5.2 Données latticielles	21
5.2.1 Librairie spdep	21
5.2.2 Critères d'adjacence et matrices de poids spatiaux	22
5.2.3 Moyenne spatiale mobile et proximité	23
5.2.4 Autocorrélation spatiale et coefficient I de Moran	24
5.3 Données géospatiales	26
6 Bibliographie	30

En statistiques spatiales, les données sont collectées en des lieux dont on a relevé la position géographique dans le but d'utiliser cette information spatiale dans la modélisation statistique. En particulier, on cherche à modéliser ce que l'expérience courante nous enseigne : deux données rapprochées géographiquement tendent à être similaires en valeur. Cette modélisation nous permettra de réaliser des prédictions spatiales ou de tester des hypothèses en intégrant explicitement cette dépendance spatiale dans les calculs. (Denis Allard, 2012)

1 Présentation du sujet

Les statistiques spatiales répondent au type de questions suivantes : étant donné un nombre limité d'échantillons de sols pour des concentrations de métaux lourds d'un ancien site industriel de Montréal, qu'elles sont les endroits les plus pollués et qu'elle est la moyenne de pollution du site ? Est-ce que la distribution du nerprun cathartique (arbuste exotique envahissant) est aléatoire dans le parc Mont-Royal, et si non est-ce que sa distribution dépend de facteurs comme l'humidité dans le sol, l'ensoleillement ? À Montréal, est-ce que la stratégie de gestion du recyclage des différentes mairies d'arrondissement est influencée par les arrondissements voisins ou est-elle indépendante ?

Fondés par des chercheurs préoccupés par des enjeux concrets forestier, minier et de l'épidémiologie, ils ont adapté les modèles statistiques classiques à des modèles tenant compte de la dépendance géographique des données. Ce vaste sous-ensemble de la statistique a explosé depuis les années 60.

Les statistiques spatiales réfèrent à l'application de concepts et méthodes statistiques à des données ayant une structure spatiale explicite en 2 ou 3 dimensions. Intuitivement, on peut comprendre qu'une plus forte corrélation existe entre des échantillons de sol (concentration de zinc par exemple) situés proche les uns des autres. La première étape des analyses statistiques est souvent d'identifier cette structure de corrélation/relation dans l'espace avant de pouvoir faire des analyses plus approfondies.

Plusieurs champs d'applications utilisent des méthodes de statistiques spatiales dans le traitement et l'analyse de leurs données: l'agriculture, la géologie, la science des sols, l'hydrologie, l'écologie, l'industrie minière, la foresterie, la qualité de l'air, la télédétection, les études sociales/économiques, l'épidémiologie .

L'effervescence de ces différents champs d'application, les différentes méthodes utilisées pour analyser les différents types de données spatiales et les enjeux liés à la manipulation des données spatiales sont à priori un défi pour le néophyte en statistiques spatiales. En effet, Bivand sur le site du CRAN répertorie plus de 185 librairies R sur les statistiques spatiales , l'un de ces librairies d'analyse spdep déploient plus de 2000 fonctions et 1600 pages de documentation.

Étant donné l'étendue du sujet traité, ce tutoriel ne va présenter qu'un aperçu des statistiques spatiales en fonction de 3 principaux types de données spatiales, de leurs méthodes et ressources respectives en R. Dans le contexte de ce tutoriel, nous n'allons pas couvrir les méthodes et librairies pour importer et exporter les objets spatiaux, ni traiter la manière de projeter ou représenter les données spatiales en 2 dimensions sur une sphère comme la Terre, ni les librairies d'aide à la visualisation et la cartographie.

2 Revue de littérature

2.1 Introduction/sources générales

L'article de Getis, Spatial Statistics peut servir d'orientation aux statistiques spatiales : de l'historique du domaine, des différentes catégories d'analyses, des enjeux et des tendances futures. Deux ouvrages clés ont recensé le domaine

des statistiques spatiales et semble être abondamment référencés dans la littérature :

- Le livre *Statistics for Spatial Data* de Cressie (1993) est un ouvrage phare du domaine et est encore d'actualité car il a su définir et catégoriser ce vaste domaine en trois branches majeures (données ponctuelles, données latticielles et données géostatistiques). Cette catégorisation est encore utilisée par plusieurs auteurs et nous la reprenons pour structurer ce tutoriel.
- Un deuxième ouvrage important et plus récent est le *Handbook of Spatial Statistics* (Gelfand, Diggle, Guttorp, Montserrat 2010) . L'ouvrage est exhaustif et tout en reprenant la structure de Cressie actualise l'état des connaissances dans chacune des branches des statistiques spatiales. De plus, l'ouvrage traite également de la perspective spatio-temporelle.

Alternativement, des notes de cours d'un cours avancé de statistiques spatiales à l'Université de Pennsylvanie (Smith 2020) sont très complètes, récentes et accessible en ligne sans droit d'accès. Encore une fois, la structure du document emprunte l'approche familière de Cressie. *Geospatial Analysis* (De Smith, Goodchild, Longley - 2018) Un guide exhaustif que l'on retrouve en version web ou en format livre , est une excellente ressource complète et à jour des objets spatiaux ainsi que des méthodes de traitements statistiques. Son traitement intégré du champ d'étude (par opposition à l'approche de catégorisation de Cressie) implique une bonne connaissance de base avant de traiter de problème précis. Finalement, on peut retrouver des articles à la fine pointe du domaine dans le journal académique *Spatial Statistics* .

2.2 Branches majeures des statistiques spatiales

Les ressources de la section précédente sont exhaustives et décrivent bien les branches majeures du domaine. Cependant, si notre question de recherche est bien identifiée ainsi que le type de données utilisées, on peut concentrer nos efforts de recherche sur la branche des statistiques spatiales la plus pertinente.

2.2.1 Type de données ponctuelles (spatial point patterns)

Pour ce type de données, la localisation des points dans une zone est l'objet de l'étude. On peut se demander si la répartition d'une espèce d'arbres dans une forêt est régulière ou présente des agrégats ? Gimond (2019) dans *Point Pattern Analysis* , fait l'inventaire des différentes méthodes d'analyses des données ponctuelles (centrographie, étude de la densité dont l'approche quadrat, analyse de la distance entre les points dont les plus proches voisins et les fonction K et L) ainsi que l'impact des effets de premier et deuxième ordre.

2.2.2 Type de données latticielles (lattice and areal unit data)

Les données latticielles représentent un nombre fini d'unité géographique (hauteur des arbres d'une forêt dans le parc Lafontaine, distribution de l'âge dans les arrondissements de Montréal, pixels dans une image). Les données qui représentent une unité sont des valeurs agrégés (moyenne de l'âge de la population). Le chapitre *Areal Data Analysis* du livre *Notebook for Spatial Data Analysis* est une description de ce type de données et donne quelques exemples de cartes.

2.2.3 Type de données géostatistiques (continuous spatial variation data)

Les données géostatistiques sont des données qui peuvent être interpolées à partir de quelques échantillons. L'article en français de Gabriel (2010) est une bonne introduction aux aspects théoriques des données géostatistiques. Ce mémoire de maîtrise (Baillargeon 2005) de l'Université Laval fait un bon recensement du krigeage, une des principales méthodes d'interpolation spatiale, alors que cette page web est une bonne introduction à cette méthode.

2.3 Des références pour quelques domaines d'application

2.3.1 Écologie

Selon Moat (2015), "Spatial Analysis: A Guide for Ecologists (Dale, Fortin, 2014) est un excellent ouvrage de référence pour un statisticien aguerri ou pour un étudiant en écologie.

2.3.2 Géosciences

Le propos de l'article Machine Learning for the Geosciences: Challenges and Opportunities (Karpatne et al, 2017) dépasse largement notre champ d'intérêt mais à l'avantage de faire une recension des sources de données et des défis. L'article fait également état du potentiel des techniques d'apprentissage automatique et inclut une longue bibliographie.

2.3.3 Épidémiologie

Selon Choi(2013) , le livre Spatial Analysis in epidemiology (Pfeiffer et al, 2008) est une excellente introduction aux spécificités des enjeux épidémiologiques liés aux analyses spatiales mais il exige un niveau de connaissance équivalent à un deuxième cycle universitaire en statistiques.

Un article de Souris et Bichaud (2011) décrit des approches d'analyse statistique de données ponctuelles bivariées à l'aide de méthode de Moran, Pearson, du plus proche voisin et de simulation Monte-Carlo.

3 Méthodes

3.1 Modèle spatial général

Les méthodes de statistiques spatiales servent à décrire, modéliser des données géo-référencées ou localisées.

La particularité principale des statistiques spatiales est que l'hypothèse d'indépendance des données n'est pas valide., Les données spatiales sont généralement corrélées spatialement surtout pour des données proches les unes des autres. L'identification de ces corrélations est souvent l'objectif initiale de l'analyse statistique.

Étant donné l'autocorrélation des données en au moins deux dimensions, les méthodes statistiques inférentielles classiques ne sont plus valables. Il faut donc des outils spécifiques permettant de tenir compte de l'autocorrélation spatiale dans nos analyses statistiques et d'éviter que celles-ci n'introduisent des biais dans l'estimation des paramètres.

Le modèle suivant décrit par Cressie (1993) est un modèle spatial général qui tient compte de la possibilité que les données soient continues ou discrètes et la position des données peut être régulières ou irrégulières.

Supposons que $s \in \mathbb{R}^d$ est une location générique dans un espace de dimension et supposons qu'une observation $Z(s)$ à cette espace est une quantité aléatoire.

$$Z(s); s \in \mathbb{D}$$

Une catégorisation reprise par de nombreux auteurs dont Diggle et Bivand, Cressie (1993) distingue trois types de données, selon la nature du domaine D, appelant des traitements statistiques spécifiques : les données ponctuelles, les données latticielles et les données géostatistiques.

3.2 Données ponctuelles (point pattern analysis)

Des données ponctuelles sont un "processus de points" (ou "point pattern") qui représente la position ou les coordonnées de différents événements au sein d'une région prédéfinie. La localisation spatiale des points est l'objet de l'étude. Par exemple, une question centrale lors de l'étude de la répartition spatiale d'une espèce d'arbres dans une forêt est de savoir si la répartition est plutôt régulière, aléatoire ou si elle présente des agrégats de points. La répartition pourra également être analysée en fonction de covariables comme l'altitude des arbres de notre forêt.

Type de distribution - Processus groupés et ordinaires D'autres types de processus de points existent et peuvent généralement être catégorisés comme groupés, dans lesquels les points auront tendance à être plus proches les uns des autres, ou ordinaires/réguliers, dans lesquels les points auront tendance à être plus espacés les uns des autres. Voici quelques méthodes pour analyser la répartition des points dans une région :

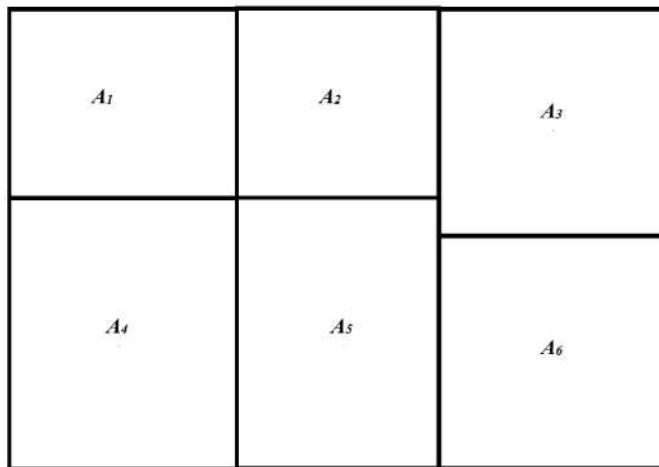
Intensité L'approche par intensité revient à calculer la densité moyenne de points dans un espace, soit l'intensité (le nombre de points divisés par l'aire de la région). Cette intensité peut ainsi être constante (uniforme) ou non. La méthode d'estimation par quadrat permet de vérifier la façon dont la densité varie à travers la région à l'étude en divisant l'espace en plusieurs sous-espaces.

Distribution des plus-proches voisins Une autre façon de déterminer le regroupement ou l'intensité d'un processus de points est de considérer chaque point et comment il se rapporte aux autres. Une mesure de cela est la distribution des distances de chaque point par rapport à son plus proche voisin. La fonction de distribution cumulée $G(r)$ correspond à la probabilité de trouver un plus proche voisin dans une distance r .

3.3 Les données latticielles (discrete spatial variation, including lattice and areal unit data)

Si les données ponctuelles sont principalement représentées comme des points dans une région à l'étude, les données latticielles correspondent à des données agrégées pour chaque partition spatiale d'une région. Les données latticielles seront principalement représentées comme des polygones, pour autant l'on cherche toujours à déterminer si des données géographiquement proches tendent à être similaires en valeur. Comme on vient à utiliser ici des polygones, l'on peut s'intéresser aux relations entre les polygones adjacents d'une région.

Critères d'adjacence et matrices de poids spatiaux L'on peut considérer que deux zones sont adjacentes si elles partagent un même côté. C'est ce qui est appelé le critère Rook, dans l'exemple ci-dessous A1 et A2 seraient adjacents mais pas A2 et A4. Le critère Queen va lui permettre l'adjacence de façon diagonale, A2 et A4 seraient donc adjacents mais toujours pas A2 et A6.



Pour pouvoir utiliser cette information plus facilement, on peut la représenter comme un vecteur de valeurs binaires qui fera office de poids pour vérifier l'adjacence de deux polygones soit par exemple pour A1, les polygones A2 et A4 sont adjacents et on donc une valeur de 1 dans le vecteur de poids w1:

- Critère Rook : $w1 = [0,1,0,1,0,0]$
- Critère Queen : $w1 = [0,1,0,1,1,0]$

Si l'on fait la même chose pour tous les polygones de la région, l'on obtient finalement une matrice de poids. Pour ce type de données spatiales, l'objectif est de déterminer si des données géographiquement proches tendent à être similaires en valeur. Comme on vient à utiliser ici des polygones, l'on peut s'intéresser aux relations entre les polygones adjacents d'une région. Voici trois méthodes utilisées pour ce type de problématique:

Moyenne spatiale mobile et proximité La moyenne spatiale mobile est une variation de la moyenne qui sera calculée en prenant en considération les valeurs des polygones adjacents au polygone considéré. La moyenne spatiale mobile nous offre déjà une première représentation du phénomène d'autocorrélation spatiale, ce qui signifie que la valeur d'une variable n'est pas indépendante de la valeur de cette même variable dans les régions voisines.

Autocorrélation spatiale et coefficient I de Moran

Le coefficient I de Moran ressemble fortement à un calcul de corrélation mais qui permet ici d'intégrer l'influence des polygones adjacents et est défini comme ceci :

$$I = \frac{n}{\sum_{i=1}^n \sum_{j=1}^n w_{ij}} \frac{\sum_{i=1}^n \sum_{j=1}^n w_{ij}(x_i - \bar{x})(x_j - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Où N est le nombre d'unités spatiales indexées par i and j; x est la variable d'intérêt; \bar{x} est la moyenne de x; w_{ij} est une matrice de poids spatiaux avec des zéros sur la diagonale (i.e., $w_{ii} = 0$); et W est la somme de tous les w_{ij} .

Bien que la formule paraîsse impressionnante, elle ressemble fortement à un calcul de corrélation mais qui permet ici d'intégrer la matrice de poids.

La valeur du coefficient I peut varier de -1 à 1, comme un coefficient de corrélation plus classique. L'importance du coefficient de Moran est qu'il permet également un test formel pour vérifier la présence d'autocorrélation spatiale.

Simulation Monte-Carlo Une simulation Monte-Carlo peut être utilisée pour ce type d'analyse. Les valeurs des variables seront assignées de façon aléatoire aux polygones et le coefficient I de Moran sera calculé. Cette procédure sera répétée n-fois afin d'établir une distribution des valeurs attendues. La valeur observée du coefficient est ensuite comparée à la distribution simulée afin de déterminer à quel point il est probable que ces valeurs soient aléatoires.

3.4 Les données géostatistiques (geostatistical - continuous spatial variation)

Les données géostatistiques sont des données dont la variable d'intérêt peut être interpolée pour tout point dans une région à partir de quelques échantillons. L'exemple typique est celui de la prospection minière, où la quantité d'un filon d'or dans une mine est interpolée à partir d'un nombre suffisant de carottes de forage.

La localisation des échantillons sont choisies pour assurer une représentation de la zone en respectant des contraintes économiques. L'intérêt des données géostatistiques est d'exploiter ces échantillons pour faire des inférences sur l'ensemble de la zone d'étude et d'ensuite estimer des intervalles de confiance .

Voici quelques exemples de questions de recherches de ce champ des statistiques spatiale : estimation de la qualité de différents minéraux dans une zone basée sur quelques carottes, l'interpolation de la pollution dans une zone à partir de quelques sites de surveillances ou même développement d'une carte topographique à partir de prélèvement d'altitude.

La seule méthode d'interpolation spatiale que nous allons décrire est le krigage qui est la première méthode à avoir tenu compte de la structure de dépendance spatiale des données. L'idée de base du krigage est de prévoir la valeur de la variable d'intérêt d'une zone qui n'a pas été échantillonné par une combinaison linéaire de données issues des zones adjacentes.

Voici les principales étapes du développement d'un modèle de krigage géostatistique :

1. Calculer la covariance des valeurs représentées à l'aide d'un semi-variogramme
2. Ajuster un modèle
3. Générer des matrices d'équation de krigage, prédire des valeurs ainsi qu'un intervalle de confiance pour chaque coordonnée dans la zone d'intérêt

Calcul du semi-variogramme (covariance des valeurs en fonction de la distance)

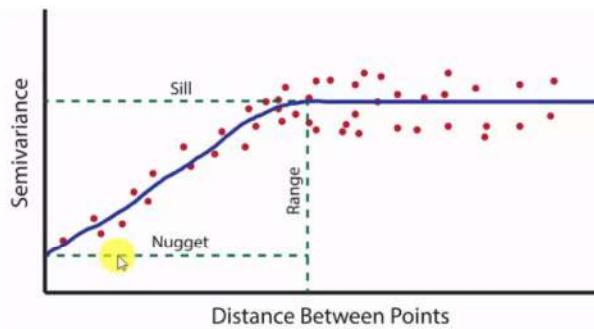
Le semi-variogramme est une fonction de dépendance spatiale. Pour calculer les points du semi-variogramme, on prend un certain nombre de paires d'échantillons (distribués en fonction de la distance euclidienne entre les pairs) et on calcule la différence au carré de la variable d'intérêt entre chaque paire.

Voici l'estimateur le plus commun du semi-variogramme, celui des moments :

$$\hat{\gamma}(r) = \frac{1}{2|N(r)|} \sum_{N(r)} (z(s_i) - z(s_j))^2$$

où $N(r) = \{(i, j) \text{ tel que } |s_i - s_j| = r\}$ et $|N(r)|$ est le nombre de paires distinctes de l'ensemble $N(r)$.

Dans la représentation d'un exemple de semi-variogramme (figure X), on se rend compte que plus les points sont rapprochés entre-eux, plus les valeurs des données d'intérêt sont corrélées (faible semi-variance). Après une certaine distance, la corrélation est faible (semi-variance élevée).



Modélisation du semi-variogramme

Plusieurs modèles sont possibles pour tenter de représenter les différences structures de covariance :

- Modèle avec palier (exponentiel, sphérique...)

- Modèle sans palier (linéaire, puissance...)

Cette modélisation est représentée par la ligne bleue dans la figure précédente.

Générer des équations de krigage

Le modèle de base du krigage possède une forme similaire au modèle de régression linéaire mais dans lequel les erreurs sont maintenant supposées dépendantes spatialement. Il s'énonce comme suit :

$$Z(s) = (\mu) + d(s), s < U + 2208 > D$$

où (μ) est la structure déterministe pour l'espérance de $Z(\mu)$. La structure de (μ) précise le type de krigage. Voici les trois grands types de krigage en ordre croissant de complexité :

- Le krigage simple : $(\mu) = m$ est une constante connue
- Le krigage ordinaire : $(\mu) = \mu_0$ est une constante inconnue.
- Le krigage universel ou dérive externe: $(\mu) = Pp_j = 0f_j(s)j$ est une combinaison linéaire de fonctions de la position s

Finalement $d(\mu)$ est une fonction aléatoire, d'espérance nulle et de structure de dépendance connue. $d(\mu)$ peut être estimé par un modèle de semi-variogramme.

4 Ressources R

4.1 Introduction

La statistique spatiale est un vaste sujet cependant l'un des auteurs d'un livre *Applied Spatial Data Analysis with R* Roger Bivand maintient une page web sur le site du CRAN avec 185 librairies liées aux statistiques spatiales . Comme ce site est exhaustif et maintenu à jour, notre conseil est de le consulter en parallèle avec la lecture de cette section. Au lieu de décrire les 185 librairies dans cette section, nous allons décrire l'organisation de chaque section du site du CRAN, identifier les 30 librairies les plus téléchargées et expliquer brièvement les principales méthodes des librairies qui seront traitées dans la section 5 du tutoriel.

4.2 Catégories principales des librairies R en statistiques spatiales

L'on peut classifier les librairies R reliées aux statistiques spatiaux en deux grandes catégories :

Les librairies utilitaires qui permettent d'importer, de structurer, de manipuler, de visualiser les données ainsi que les résultats des analyses. La seconde catégorie correspond aux librairies qui implémentent les analyses statistiques propres aux statistiques spatiales. Certaines librairies sont plus petites et très spécialisées avec seulement quelques fonctions alors que d'autres comme spatstat a plus de 2000 fonctions répertoriées et 1600 pages de documentation.

Les 185 librairies de R sont catégorisées avec la structure suivante:

Catégories	Descriptions	Librairies
Classes for spatial data and metadata	Classes partagées pour l'entreposage et le traitement et la visualisation des données spatiales	
Spatial data - general	Classes de données spatiales principalement de type vecteur	sp, sf, stars, stplanr, spacetime, inlmisc, maptools
Raster data	Classes de données spatiales de type matricielles	raster, stars, spatial.tools
Geographic metadata	Classes et méthodes pour lajout et le traitement de métadonnées selon différents standards	geometa, ncdf4
Reading and writing spatial data	Chargement et écriture des données spatiales	
Reading and writing spatial data	rgdal est la librairie principale qui facilite l'importation et l'exportation de cartes de format vecteurs et matricielles selon différents standards dont GDAL, OGR, OGC ESRI	rgdal, vec2dtransf
Reading and writing spatial data - data formats	Autres librairies plus spécialisées qui donnent accès à des formats OGS, ESRI...	sf, regeo, wkb, geojson, geojsonio, geoaxe lawn, maps, geometa, ows4R, ncdf4, geometa, mapdata, mapproj, shapefiles, maptools, gmt
Reading and writing spatial data - GIS Software connectors	Librairies permettant de se connecter avec les principaux logiciels GIS	rpostgis, RPostgreSQL, postGISTools, rgrass7, spgrass6, RSAGA, RQGIS, RPvGeo
Interfaces to Spatial Web-Services	Librairies pour faciliter la connexion à des services/outils web	ows4R, geosapi, geonapi,
Specific geospatial data sources of interest	Répertoires de cartes de toutes sortes (frontières des pays, éléments topographique (routes, lacs...), recensements américains...)	naturalearth, rworldmap, rworldxtra, cshapes, marmap, maptools, rgbf, geonames, OpenStreetMap, osmar, tidyicensus, tigris
Handling spatial data	Traitement et manipulation des données spatiales	
Data processing - general	Librairies utilitaires pour des données vectorielles (échantillonnage, calcul des distance, segmentation des données...)	rgdal, maptools, rgeos, raster, gdalUtils, gdistance, cshapes, geosphere, spsurvey, trip, spcosa, magclass, taRifx, geoaxe, lawn, areal, qualmap
Data processing - raster and imagery data	Exploration et traitement de données matricielles pour la télédétection	landsat
Data cleaning	Inspection et traitement des données spatiales pour gérer les erreurs topologiques	cleango, lwgeom
Visualizing spatial data	Visualisation des données spatiales	
Base visualization packages	Les librairies de classes de bases (sp, sf, raster) intègrent également des méthodes de visualisation qui peuvent être améliorées avec des librairies de palettes de couleurs	sp, sf, raster, rasterViz, RColorBrewer, viridis, classInt
Thematic cartography packages	Outils pour rapidement visualiser des cartes à laide de modèles	tmap, quickmapr, cartography, mapmisc, PBSmapping, PBSmodelling, GEOmap, geomapdata
Packages based on web-mapping frameworks	Outils pour rapidement visualiser des cartes à déployer sur le web à laide de modèles	mapview, leaflet, leafletR, RgoogleMaps, plotKML, ggmap, mapedit, ggsn
Building Cartograms	Librairies de cartogrammes (substitution d'une variable comme population au lieu de laire d'un pays)	micromap, recmap, statebins, cartogram, geogrid
Analyzing spatial data	Librairie pour analyser les données spatiales (univarié, multivarié, interpolation, structure de corrélation...)	
Point pattern analysis	Outils d'analyses pour identifier la nature de la distribution des données et la répartition d'objets dans une région (arbres...)	spatial, spatstat, spatgraphs, splancs, smacpod, ecespa, ads, aspace, ash, DStat, dbmss, spatialsegregation, latticeDensity
Geostatistics	Les librairies géostatistiques permettent l'analyse déchantillons afin de réaliser des inférences sur l'ensemble de la zone étudiée et destimer des intervalles de confiance	gstat, geoR, varidag, automap, intampl, fields, spatial, spBayes, ramps, geospt, spsann, geostatsp, FRK, RandomFields, CompRandRld, constrainedKriging, geospt, spTimer, rtop, georob, SpatialTools, serraest, spn, ExceedanceTools, deldir, tripack, akima, MBA, spatialCovariance, tgp, Stem, FieldSim, SSN, ipdw, Rsurvey
Disease mapping and areal data analysis	Ces librairies traitent des liens entre différents sites représentant chacun une zone de données agrégées	DCluster, spdep, spatialreg, spmoran, SpatialEpi, diseasemapping, AMOEBA, seg, OasisR, spgrw, GWmodel, sparr, CARBayes, spaMM, PReMiM, spatsur, spBayesSurv, spselect
Spatial regression	Librairies spécialisées dans la régression spatiale avec différentes distributions (point pattern ou géostatistiques)	nlme, spatialreg, spdep, sphet, McSpatial, S2sls, spanel, splm, spatialprobit, ProbitSpatial, starma
Ecological analysis	Librairies spécialisées dans l'analyse des données environnementales.	ade4, adehabitatHR, adehabitatHS, adehabitatLT, adeabitatMA, pastecs, vegan, tripEstimation, trip.ncf, spind, rangeMapper, siplab, ModelMap, SpatialPosition, Watersheds, Récrus, ngsptial, landscapemetrics, FGRASTATS

4.3 Les 30 librairies en statistiques spatiales les plus téléchargées

A l'aide d'un peu de web scraping, nous avons identifié parmi les 185 librairies répertoriées, les 30 plus populaires téléchargées récemment. Il faut noter que certaines librairies comme RColorBrewer, viridis et igraph sont utilisées dans d'autres domaine que l'analyse spatiale et que plusieurs librairies utilitaires peuvent être des dépendances d'autres librairies. Nonobstant ces limites ce tableau donne une représentation des librairies le plus importantes dans le domaine.

Les 30 librairies téléchargées sur CRAN depuis 1 mois (au 27 février 2020) :

	Nom de la librairie	% de téléchargement	Description sommaire	Sous-catégorie
1	RColorBrewer	14.50%	color schemes for maps and other graphics	Base visualization
2	sp	8.30%	classes and methods for dealing with spatial data	Spatial data - general
3	viridis	7.70%	color schemes for maps and other graphics	Base visualization
4	igraph	6.40%	Routines for simple graphs and network analysis. Conversion	Spatial data - general
5	maptools	5.70%	functions between spatstats ans sp classes	Spatial data - general
6	nlme	4.30%	Spatial regression (geostatistical)	Spatial regression
7	rgdal	4.20%	Provides binding to GDAL and PROJ.4 map libraries as well as writing raster and vector files	Reading and writing spatial data
8	classInt	4.00%	Functions for choosing class intervals for thematic cartography.	Base visualization
9	sf	3.80%	Support for simple features, a standardized way to encode spatial vector data	Spatial data - general
10	raster	3.60%	Reading, writing, manipulating, analyzing and modeling of gridded spatial data.	Raster data
11	RgoogleMaps	2.30%	Accessing Google Maps	Web-mapping framework
12	maps	2.20%	Display of maps	Spatial data - general
13	ggmap	2.10%	Spatial visualisation with Google Maps and OpenStreetMap	Web-mapping framework
14	leaflet	2.00%	Methods to view spatial objects interactively	Web-mapping framework
15	rgeos	1.70%	Interface to topology functions for sp objects using GEOS .	Data formats
16	RPostgreSQL	1.70%	Interface R with a 'PostGIS'-enabled database	GIS software connectors

17	deldir	1.30%	Calculates the Delaunay triangulation and the Dirichlet or Voronoi tessellation	Geospatial analysis
18	vegan	1.30%	Ordination methods and other useful functions for community and vegetation ecologists, For curve, surface and function fitting with an emphasis on splines, spatial data, geostatistics, and spatial statistics.	Ecological analysis
19	fields	1.30%	fitting with an emphasis on splines, spatial data, geostatistics, and spatial statistics.	Geospatial analysis
20	ade4	1.30%	Analysis of Ecological Data: Exploratory and Euclidean Methods in Environmental Sciences	Ecological analysis
21	mapproj	1.20%	Access ESRI proprietary Data formats	Data formats
22	geosphere	1.10%	Computations of distance and area to be carried out on spatial data in geographical coordinates	Data processing
23	lwgeom	0.90%	handling and reporting of topology errors and geometry validity issues basic functions for building	Data Cleaning
24	spdep	0.90%	neighbour lists and spatial weights, tests for spatial autocorrelation for areal data like Moran's I. Comprehensive open-source toolbox	Disease mapping and analysis
25	spatstat	0.90%	for analysing Spatial Point Patterns. Focused mainly on two-dimensional point patterns, PPP classes	Point pattern analysis
26	tmap	0.80%	Modern basis for thematic mapping optionally using a Grammar of Graphics syntax.	Thematic cartography
27	gstat	0.70%	Wide range of functions for univariate and multivariate geostatistics	Geostatistics
28	spacetime	0.70%	Extends the shared classes defined in sp for spatio-temporal data	Spatial data - general
29	mapview	0.60%	Methods to view spatial objects interactively, usually on a web mapping base.	Web-mapping framework
30	ncdf4	0.50%	Read and write functions for handling metadata (CF conventions) in the self-described NetCDF format.	Geographic metadata

4.4 Quelques références R incontournables en analyse spatiale

L'ouvrage Applied Spatial Data Analysis with R traite de données ponctuelles (spatial point pattern) au chapitre 7, dans les chapitres 9 à 11 des données latticielles (lattice data ou areal data) et finalement au chapitre 8 des données géospatiales.

Geocomputation with R est une ressource en ligne exhaustive sur l'analyse géographique, la visualisation et la modélisation, qui accompagne un livre du même titre.

Finalement, Datacamp offre une formation en données spatiales en R (Skill Track: Spatial Data with R) avec les cours suivants : Visualizing Geospatial Data in R, Spatial Analysis with sf and raster in R, Spatial Statistics in R, Interactive Maps with leaflet in R.

4.5 Librairies abordées durant la section tutoriel

Librairies de classes de données spatiales : SP et SF

La représentation vectorielle des données en R n'est pas forcément adaptée pour représenter des données spatiales. Pour résoudre ce problème, des classes de données spatiales furent développées pour faciliter la manipulation des données spatiales dans des objets qui ressemblent aux data frames plus classiques. Les trois principales librairies de classes sont SP, SF et Raster.

Librairie SP

Le développement de la librairie de classes et méthodes spatiales sp a débuté en 2005. Les classes sp sont toujours les plus fréquemment utilisées en R pour représenter les vecteurs et matrices spatiales.

Une des classes de base offerte dans sp est un objet spatial nommé SpatialPoints. Cet objet Spatial est une classe qui définit le système de coordonnées de référence (proj4string) ainsi que l'aire de l'objet (bbox). Les données spatiales sont stockées dans des slots auxquels il est possible d'accéder avec le symbole @ dont l'usage est similaire au symbole du dollar pour les attributs d'un data frame (*objet@slot*).

Il existe 10 sous-classes dans sp pour représenter divers types d'objets dont SpatialPoints, SpatialLines, SpatialPolygons, SpatialGrid. Les sous-classes de Spatial Points peuvent être intégrées dans un SpatialDataFrame qui est similaire au DataFrame de base de R. Ainsi, la plupart des méthodes de R fonctionne sur ces objets. Par contre, ces SpatialDataFrame ne peuvent être manipulés par des librairies hors R Base telles que dplyr et ne peuvent être visualisés avec ggplot2. La visualisation des objets sp se fait avec la fonction plot de base et une méthode de sp, spplot.

Librairie SF

La librairie sf est l'implantation en R du standard ouvert simple features pour la représentation géographique vectorielle. Cette librairie est développée très activement depuis 2016. L'objectif de sf est de remplacer à terme sp. Non seulement, l'utilisation du standard ouvert permet une meilleure intégration avec les outils logiciels SIG (GIS) tel que PostGIS, GeoJSON et ArcGIS mais il fonctionne également dans l'univers tidyverse (dplyr, ggplot2,). De plus, il est capable d'interfacer avec les librairies GDAL, GEOS et PROJ ce qui évite l'utilisation de la librairie rgdal.

Les objets sf peuvent être composés de variables correspondantes à des données spatiales ou non-spatiales. Les variables spatiales appartiennent à une classe géométrique nommée sfc. ggplot2 permet également la visualisation d'objet sf avec *geom_sf()*.

Analyses de données ponctuelles : Librairie spatstat

La librairie spatstat est une librairie open source spécialisée dans l'analyse de processus de points, principalement les processus de points en deux dimensions. La librairie gère également des processus en trois dimensions et d'autres types d'objets géométriques.

Il contient plus de 2000 fonctions pour tracer des données spatiales, réaliser des analyses exploratoires, des simulations, effectuer un échantillonnage spatial ou encore des diagnostics de modèles et de l'inférence formelle.

Les méthodes exploratoires comprennent les tests du quadrant, les fonctions K et leurs enveloppes de simulation, les statistiques de distance et des plus proches voisins. Ces méthodes seront principalement utilisées afin de comprendre la distribution du processus de points auquel l'on fait face.

Contrairement à d'autres librairies, travailler avec des processus de points signifie que l'on peut se baser sur des données dans un format plus classique, il n'est pas nécessaire de forcément travailler avec des données dans un format spatial.

L'objectif des méthodes aléatoires est généralement de tester le cas d'une distribution aléatoire complète.

Les processus ponctuels de Poisson sont les modèles les plus simples pour les modèles de points, un modèle de Poisson suppose que les points sont stochastiquement indépendants. Il peut permettre aux points d'avoir une densité spatiale non uniforme mais le cas particulier d'un processus de Poisson avec une densité spatiale uniforme est souvent appelé aléatoire spatiale complète.

Les processus ponctuels de Poisson sont inclus dans la classe plus générale des modèles de processus ponctuels de Gibbs. Dans un modèle de Gibbs, il existe une interaction ou une dépendance entre les points. De nombreux types d'interaction différents peuvent être spécifiés.

Analyse de données latticielles: Librairie spdep

La librairie spdep est une collection de fonctions pour créer des matrices de poids spatiaux à partir de polygones ou de processus de points selon les distances ou tessellations.

Cette librairie peut toujours prendre en entrée des processus de points mais va principalement nécessiter des données spatiales. Ces dernières sont généralement décrites par un champ nommé `z` qui va permettre d'ajuster des polygones sur lesquels pourront être effectuées d'autres analyses.

La librairie rend ensuite possible l'utilisation de ces objets dans des analyses spatiales, ces analyses peuvent notamment être :

- Aggrégation par région
- Tests d'autocorrélation spatiale, Morans I et Gearys Ces
- Estimés empiriques Bayesien
- Estimateurs locaux LOSH d'hétéroscédasticité spatiale

Le calcul de matrices de poids spatiaux reste pour autant au cœur du librairie et sera utile à un grand nombre des analyses disponibles. Ces dernières peuvent également être calculées de différentes façons et permettent notamment de considérer différents critères d'adjacence.

La librairie implémente également des analyses un peu plus simples et facilite notamment le calcul du moyenne spatiale mobile. Des fonctions de visualisations existent afin d'ajuster graphiquement certaines des analyses disponibles mais ce n'est pas le cas pour la majorité d'entre elles, c'est le cas de la moyenne spatiale mobile, une autre librairie de visualisation est nécessaire notamment pour visualiser les représentations géométriques de nos données.

Un avantage particulier de spdep est également que la librairie inclut des simulations de Monte-Carlo dans certains des tests disponibles, c'est le cas notamment du coefficient I de Moran. Cette procédure sera donc répétée n-fois afin d'établir une distribution des valeurs attendues. La valeur observée du coefficient est ensuite comparée à la distribution simulée afin de déterminer à quel point il est probable que ces valeurs soient aléatoires.

Analyse de données géostatistiques : Librairie gstat

Le librairie gstat offre une panoplie de fonctionnalités pour interpoler des données géostatistiques : plusieurs types de variogrammes, des analyses de krigeage ordinaires et universelles à plusieurs variables (cokriging), des analyses multivariées. Les principales fonctions sont les suivantes :

- Variogram prend en entrée les coordonnées des échantillons de la variable d'intérêt pour représenter la corrélation entre les échantillons en fonction de leur distance.
- fit.variogram modélise la structure de corrélation spatiale à l'aide du résultat issu de variogram. gstat peut estimer automatiquement cette structure ou l'usager peut le faire manuellement
- krige, fait l'interpolation et estime la variance de la variable d'intérêt sur l'ensemble de la région à l'aide du résultat de fit.variogram. Plusieurs paramètres sont possibles en fonction du type de krigeage : ordinaire, universel à une variable, universel à plusieurs variables.
-

5 Exemples d'analyses avec des données

5.1 Processus de points

Cette information spatiale peut prendre différentes formes et représenter de nombreux phénomènes. Un "processus de points" (ou "point pattern") va nous permettre de représenter la position ou les coordonnées de différents événements au sein d'une région pré-définie. Le nombre de points ainsi que leur position est considéré comme aléatoire. Une région contient un nombre infini de points soit les coordonnées (x_i, y_i) sur un plan. Le nombre de points est infini puisque n'importe quelle position qui peut être définie comme un ensemble de coordonnées contenues dans la région en est un point.

5.1.1 Librairie Spatstat

La première librairie abordée dans ce tutoriel sera la librairie "Spatstat" dont les fonctionnalités principales seront abordées un peu plus loin. Les premiers exemples présentés ci-dessous sont tous disponibles à travers la librairie et peuvent être appelés de cette façon:

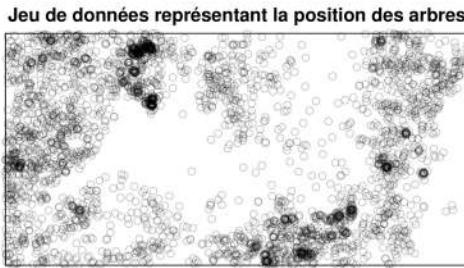
```
> data("nom du jeu de données")
```

Spatstat : <https://CRAN.R-project.org/package=spatstat>

Quick Reference Guide : <https://spatstat.org/resources/spatstatQuickref.pdf>

```
> # Install the spatstat package
> if(!require(spatstat)) install.packages("spatstat", repos = "http://cran.us.r-project.org")
> # Load the spatstat package
> library(spatstat)
> data(bci)
```

```
> par(mfrow = c(1,1))
> plot(bei, main = "Jeu de données représentant la position des arbres")
```

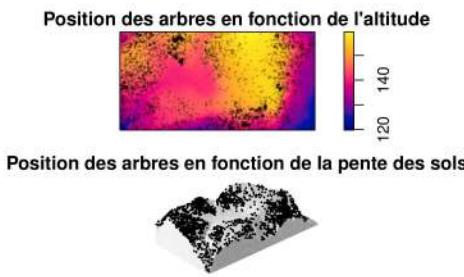


Ces données représentent la position de 3605 arbres dans une forêt tropicale. L'objet à l'étude est donc ici les coordonnées respectives de chacune des cellules.

5.1.2 Covariables

Maintenant que la partie "réponse" de notre étude est identifiée, soit les paramètres de la distribution des points, il est également possible de considérer une partie de nos données comme explicatives. Ces covariables peuvent être une fonction $Z(u)$ définie en tout point de la région observée et qui pourrait représenter par exemple l'altitude ou la pente des sols observés. Tel que :

```
> par(mfrow = c(2,1))
> plot(bei.extra$elev, main="Position des arbres en fonction de l'altitude")
> plot(bei, add=TRUE, pch=16, cex=0.3)
> M <- persp(bei.extra$elev, theta=-45, phi=18, expand=7, border=NA, apron=TRUE
+ , shade=0.3, box=FALSE, visible=TRUE
+ , main="Position des arbres en fonction de la pente des sols")
> perspPoints(bei, Z=bei.extra$elev, M=M, pch=16, cex=0.3)
```



5.1.3 Intensité

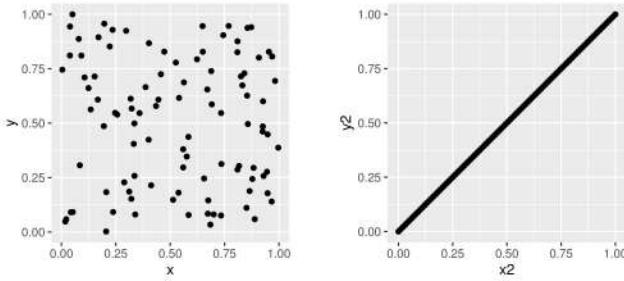
L'un des principaux objectifs de ces processus de points est de déterminer si leur distribution est aléatoire ou non. Cela qui revient à calculer la densité moyenne de points dans un espace, soit l'intensité (le nombre de points divisés par l'aire de la région). Cette intensité peut ainsi être constante (uniforme) ou non.

La librairie de visualisation *ggplot2* va nous permettre de mieux représenter ce concept.

```
> # Install the ggplot2 package
> if(!require(ggplot2)) install.packages("ggplot2", repos = "http://cran.us.r-project.org")
> if(!require(gridExtra)) install.packages("gridExtra", repos = "http://cran.us.r-project.org")
```

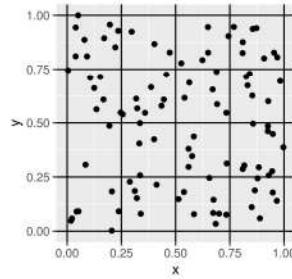
```
> # Load the ggplot2 package
> library(ggplot2)
> library(gridExtra)

> x <- runif(100, 0, 1)
> y <- runif(100, 0, 1)
> unif <- cbind.data.frame(x,y)
> x2 <- seq(0,1, length.out = 100)
> y2 <- seq(0,1, length.out = 100)
> ex2 <- cbind.data.frame(x2,y2)
> plot1 <- ggplot() +
+   geom_point(data = unif, aes(x = x, y = y)) +
+   coord_fixed()
> plot2 <- ggplot() +
+   geom_point(data = ex2, aes(x = x2, y = y2)) +
+   coord_fixed()
> grid.arrange(plot1, plot2, nrow=1, ncol=2)
```



Chacun de ces groupes est composés de 60 points, calculer l'intensité globale de chacun des ces groupes retrouvera ainsi le même résultat bien que leur distribution est très différente. Il serait possible de vérifier la façon dont la densité varie à travers la région à l'étude en divisant l'espace en plusieurs sous-espaces. Ci-dessous, 16 sous-régions sont définies et 60 points aléatoires issus d'une distribution uniforme sont représentés à l'aide de ggplot2.

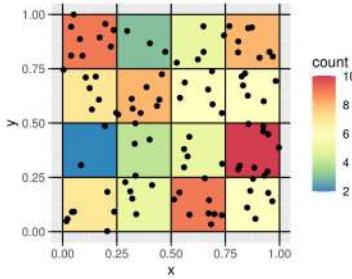
```
> ggplot() +
+   geom_vline(xintercept = seq(from = 0, to = 1, by = 0.25)) +
+   geom_hline(yintercept = seq(from = 0, to = 1, by = 0.25)) +
+   geom_point(data = unif,
+             aes(x = x, y = y)) +
+   coord_fixed()
```



Si l'on calcule l'intensité respective de chacun de ces nouveaux quadrants, l'on retrouve des résultats bien plus

représentatifs de la distribution des points. Cela est possible à l'aide de ggplot2 de cette façon:

```
> ggplot(data = unif, aes(x = x, y = y)) +
+   geom_bin2d(binwidth = c(0.25, 0.25)) +
+   geom_vline(xintercept = seq(from = 0, to = 1, by = 0.25)) +
+   geom_hline(yintercept = seq(from = 0, to = 1, by = 0.25)) +
+   geom_point() +
+   scale_fill_distiller(palette = "Spectral") +
+   coord_fixed()
```



Une approche plus formelle de ce test peut facilement être calculée à l'aide de la librairie spatstat et de sa fonction:

```
> quadrat.test(ppp)
```

Pour autant, la majorité des fonctionnalités de spatstat vont nécessiter un nouveau type d'objet en entrée, un "Planar Point Pattern" ou "ppp".

<https://www.rdocumentation.org/packages/spatstat/versions/1.63-2/topics/quadrat.test>

5.1.4 Planar Point Pattern

Un objet "ppp" vient simplement représenter un processus de points dans un plan en deux dimensions. Et sera créé de cette façon :

```
> ppp(x,y, '', window)
```

Où x est un vecteur de coordonnées x, y est un vecteur de coordonnées y et window correspond aux délimitations de la région à l'étude telle que présenté plus haut.

<https://www.rdocumentation.org/packages/spatstat/versions/1.63-2/topics/ppp>

L'utilisation d'objets "ppp" va nous permettre d'analyser des processus de points dont le comportement est bien différent d'une loi uniforme comme dans l'exemple sur l'intensité ci-dessus.

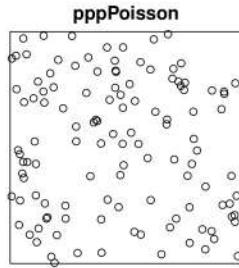
5.1.5 Processus de points de Poisson

Un processus de points de Poisson va nous permettre de générer des points aléatoires sur la base d'un paramètre lambda qui correspond à l'intensité espérée des points dans la région à l'étude, grâce à la fonction:

```
> rpoispp(lambda, win)
```

<https://www.rdocumentation.org/packages/spatstat/versions/1.63-2/topics/rpoispp>

```
> # La fonction spatstat::owin(xrange= , yrange= ) nous permet de facilement générer une région
> # qui prend comme paramètres les coordonnées limites de x et de y
> # https://www.rdocumentation.org/packages/spatstat/versions/1.63-2/topics/owin
> region <- owin(xrange=c(0,1), yrange=c(0,1))
> # Définition du paramètre d'intensité pour le processus de poisson (Soit le nombre de points divisés par
> # Soit ici 100 points divisé par l'aire d'un cercle de rayon 5
> lambda <- 100 / area(region)
> # L'on peut maintenant simplement appeler la fonction ci-dessus
> pppPoisson <- rpoispp(lambda, region, lmax = 100)
> plot(pppPoisson)
```

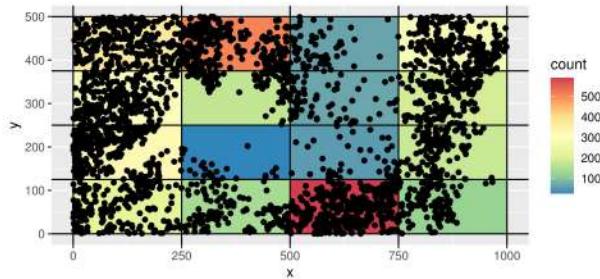


5.1.6 Processus groupés et ordinaires

D'autres types de processus de points existent et peuvent généralement être catégorisés comme groupés, dans lesquels les points auront tendance à être plus proches les uns des autres, ou ordinaires/réguliers, dans lesquels les points auront tendance à être plus espacés les uns des autres. Le processus de poisson présenté plus haut ne correspond pour autant à aucune de ces deux catégories puisqu'il que son intensité est uniforme, la répartition des points ne dépend pas de la position des autres.

Un processus de points de Thomas, un processus groupé, permet de représenter des points qui ont tendance à former des clusters, cela peut être notamment intéressant pour représenter des arbres qui apparaissent proches les uns des autres. Ce qui correspond aux données de notre premier exemple de ce tutoriel (les données représentant la position de 3605 arbres dans une forêt tropicale), en appliquant ce que l'on a appris sur l'intensité, il est possible de facilement visualiser cela.

```
> clusterArbres <- as.data.frame.ppp(bei)
> ggplot(data = clusterArbres, aes(x = x, y = y)) +
+   geom_bin2d(binwidth = c(250, 125)) +
+   geom_vline(xintercept = seq(from = 0, to = 1000, by = 250)) +
+   geom_hline(yintercept = seq(from = 0, to = 500, by = 125)) +
+   geom_point() +
+   scale_fill_distiller(palette = "Spectral") +
+   coord_fixed()
```



Un tel processus peut-être facilement généré à l'aide de la fonction:

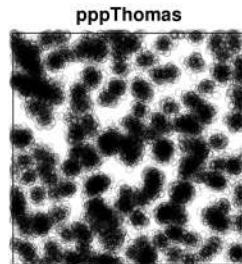
```
> rThomas(kappa, scale, mu, win)
```

Cette fonction va prendre deux paramètres de plus que la fonction pour les processus de Poisson (kappa "correspond" ici au lambda de la fonction précédente) soit scale et mu.

1. scale: Écart-type d'un point par rapport au centre du cluster auquel il appartient
 2. mu: Nombre moyen de points ou intensité moyenne dans un cluster
-

<https://www.rdocumentation.org/packages/spatstat/versions/1.63-2/topics/rThomas>

```
> region <- owin(xrange=c(0,40), yrange=c(0,40))
> kappa <- 200 / area(region)
> # L'on peut maintenant simplement appeler la fonction ci-dessus
> pppThomas <- rThomas(kappa, scale = 0.7, mu = 70, win = region)
> plot(pppThomas)
```



5.1.7 Distribution des plus-proches voisins

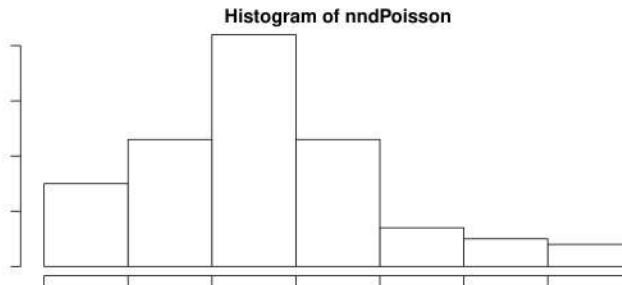
Une autre façon de déterminer le regroupement ou l'intensité d'un processus de points est de considérer chaque point et comment il se rapporte aux autres. Une mesure de cela est la distribution des distances de chaque point par rapport à son plus proche voisin.

La fonction `nndist()` prend en entrée un processus de point et retourne pour chaque sa distance avec son plus proche voisin.

```
> nndist(ppp)
```

<https://www.rdocumentation.org/packages/spatstat/versions/1.63-3/topics/nndist>

```
> #Calcul de la distribution pour le processus de points uniformes de poisson créé plus tôt
> nndPoisson <- nndist(pppPoisson)
> #Histogramme des distances calculées
> hist(nndPoisson)
```



Plutôt que de travailler avec la densité des plus proches voisins comme représentée dans cet histogramme, il est possible d'utiliser la fonction de distribution cumulée $G(r)$, qui correspond à la probabilité de trouver un plus proche voisin dans une distance r .

Il est possible d'estimer empiriquement G avec la fonction `Gest` ou `ppp` est un processus de points :

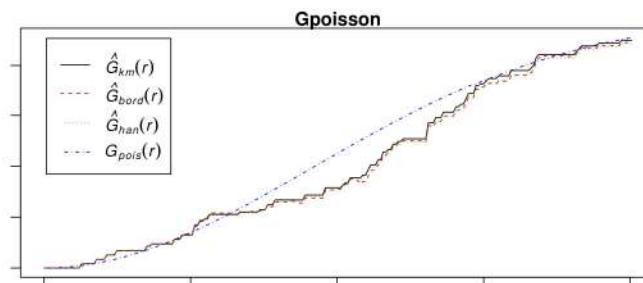
```
> Gest(ppp)
```

De plus, comme pour le test du quadrant, l'on va pouvoir comparer cette probabilité observée “ G ” avec la probabilité théorique d'une distribution uniforme de Poisson que l'on peut calculer numériquement :

```
> G(r) = 1 - exp(-lambda * pi * r ^ 2)
```

<https://www.rdocumentation.org/packages/spatstat/versions/1.63-3/topics/Gest>

```
> #par(mfrow = c(3,1))
> # Estimer G(r) pour un processus de poisson
> Gpoisson <- Gest(pppPoisson)
> # Graphique de G(r) vs. r
> plot(Gpoisson)
> # Même chose pour le processus de Thomas
> nndThomas <- nndist(pppThomas)
> hist(nndThomas)
> Gthomas <- Gest(pppThomas)
> plot(Gthomas)
```



5.2 Données laticielles

Dans les premières parties de ce tutoriel, grâce à des processus de points l'on a pu déterminer des relations spatiales entre différents événements en fonction soit de leur position relative soit en fonction de leur distance les uns des autres. D'où notamment les techniques du quadrant ou de distances vu plus haut.

Pour autant, en travaillant avec des données spatiales et plus seulement des points, l'on peut représenter ces relations de nombreuses façons.

5.2.1 Librairie spdep

La prochaine librairie abordée dans ce tutoriel sera la librairie "spdep" dont les fonctionnalités principales seront abordées un peu plus loin. Les exemples présentés ci-dessous sont tous disponibles à travers la librairie `geodaData` et peuvent être appelés de cette façon:

```
> data("nom du jeu de données")
> if(!require(spdep)) install.packages("spdep", repos = "http://cran.us.r-project.org")
> if(!require(remotes)) install.packages("remotes", repos = "http://cran.us.r-project.org")
> remotes::install_github("spatialanalysis/geodaData")
> library(spdep)
> library(geodaData)
> data("ohio_lung")
```

Les données importées contenues dans le dataset `ohio_lung` sont des données représentant le nombre de mort par cancer des poumons pour chaque comté de l'état de l'Ohio en 1968, 1978 et 1988. Ces données contiennent des données spatiales et l'on peut ainsi créer un nouvel objet pour les contenir avec la fonction:

```
> as(dataset, 'Spatial')
> ohio_lung.sp <- as(ohio_lung, 'Spatial')
> plot(ohio_lung.sp)
```



L'on obtient ainsi une visualisation bien différente de celles dont on peut avoir l'habitude. Des polygones représentant l'état et les comtés sont automatiquement générés.

Malgré que l'on utilise maintenant des données spatiales, l'objectif des statistiques spatiales reste le même, l'on cherche toujours à déterminer si des données géographiquement proches tendent à être similaires en valeur. Comme on vient à utiliser ici des polygones, l'on peut s'intéresser aux relations entre les polygones adjacents d'une région.

5.2.2 Critères d'adjacence et matrices de poids spatiaux

Les critères d'adjacences sont abordés et expliqués plus en détails dans la section 3.3 de ce document. Leur implémentation en R sera présentée ici:

La librairie `spdep` va nous permettre de facilement calculer des matrices de poids et d'identifier les régions adjacentes à l'aide des fonctions :

```
> poly2nb(pl, queen=TRUE)
```

Qui va nous permettre d'identifier les polygones adjacents et qui utilise par défaut le critère Queen. Ainsi que :

```
> nb2listw(neighbours)
```

Qui va calculer la matrice de poids et l'enregistrer dans une liste et qui prendre en entrée le résultat de la fonction `poly2nb`.

<https://www.rdocumentation.org/packages/spdep/versions/1.1-3/topics/poly2nb> <https://www.rdocumentation.org/packages/spdep/versions/1.1-3/topics/nb2listw>

```
> ohio_lung.nb <- poly2nb(pl = ohio_lung.sp, queen = TRUE)
> summary(ohio_lung.nb)
```

Neighbour list object:

Number of regions: 88

Number of nonzero links: 462

Percentage nonzero weights: 5.965909

Average number of links: 5.25

Link number distribution:

3 4 5 6 7 8

12 14 18 29 14 1

12 least connected regions:

2 4 6 12 19 62 63 77 81 86 87 88 with 3 links

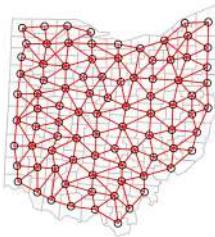
1 most connected region:

29 with 8 links

```
> ohio_lung.w <- nb2listw(ohio_lung.nb)
```

L'on a donc identifié tous les comtés adjacents sur notre carte de l'Ohio.

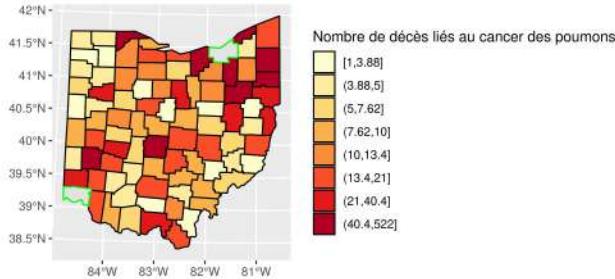
```
> plot(ohio_lung.sp, border = "gray")
> plot(ohio_lung.nb, coordinates(ohio_lung.sp), col = "red", add = TRUE)
```



5.2.3 Moyenne spatiale mobile et proximité

La moyenne spatiale mobile est une variation de la moyenne qui sera calculée en prenant en considération la matrice de poids que l'on a calculée plus tôt. Si l'on représente pour le moment le nombre de décès liés au cancer des poumons sur la carte que nous avons générée plus tôt, l'on peut remarquer certaines zones qui regroupent le plus de cas.

```
> ggplot(data = ohio_lung) +
+   geom_sf(aes(fill = cut_number(LM68, 8), ),
+           color = "black") +
+   geom_sf(data = subset(ohio_lung, COUNTYID == 31),
+           color = "green") +
+   geom_sf(data = subset(ohio_lung, COUNTYID == 18),
+           color = "green") +
+   scale_fill_brewer(palette = "YlOrRd") +
+   labs(fill = "Nombre de décès liés au cancer des poumons") +
+   coord_sf()
```



L'on peut remarquer que la majorité des cas se regroupent en deux zones au sud-ouest et au nord-ouest de l'état. L'on peut identifier les deux comtés avec le plus grand nombre de cas en les entourant en vert, cela nous permettra d'identifier l'effet qu'aura notre prochaine transformation sur ces deux zones.

La fonction `lag.listw` va nous permettre de calculer la moyenne spatiale mobile et prends ces paramètres :

```
> lag.listw(weightMatrix, var)
> LM68.sma <- lag.listw(x = ohio_lung.w, ohio_lung$LM68)
> ohio_lung$LM68.sma <- LM68.sma
```

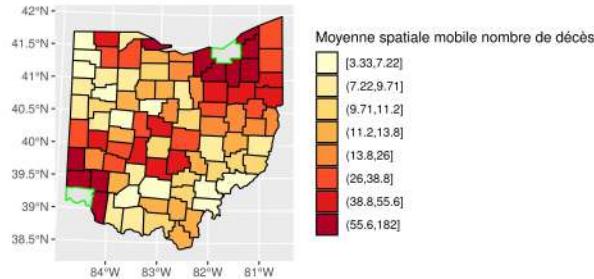
L'on peut maintenant reproduire la carte ajustée plus tôt mais en replaçant cette fois-ci le nombre de décès par la moyenne mobile que l'on vient de calculer.

```
> ggplot() +
+   geom_sf(data = ohio_lung,
```

```

+         aes(fill = cut_number(LM68.sma, 8)),
+         color = "black") +
+     geom_sf(data = subset(ohio_lung, COUNTYID == 31),
+             color = "green") +
+     geom_sf(data = subset(ohio_lung, COUNTYID == 18),
+             color = "green") +
+     scale_fill_brewer(palette = "YlOrRd") +
+     labs(fill = "Moyenne spatiale mobile nombre de décès") +
+     coord_sf()

```



L'on remarque ainsi que ces nouvelles valeurs évoluent fortement, les deux comtés qui regroupaient le plus de cas influencent très fortement les comtés voisins. Alors que ces deux comtés présentent maintenant des valeurs plus faibles qu'auparavant, bien que toujours élevées. L'on pourrait faire l'hypothèse de la présence d'un facteur géographique qui influencerait le nombre de cas dans cette zone. L'on peut clairement voir une propagation en cercles concentriques autour du comté identifié au nord-est.

5.2.4 Autocorrélation spatiale et coefficient I de Moran

Comme nous venons de le voir, la moyenne spatiale mobile nous offre déjà une première représentation du phénomène d'autocorrélation spatiale, ce qui signifie que la valeur d'une variable n'est pas indépendante de la valeur de cette même variable dans les régions voisines.

Le coefficient I de Moran est abordé et expliqué plus en détail dans la section 3.3 de ce document. Son implémentation en R sera présentée ici :

Le package **spdep** va de plus nous permettre de calculer très facilement le coefficient grâce à la fonction “**moran**” qui prend comme paramètres :

```
> moran(x, listw, n, S0)
```

Où **x** est un vecteur numérique, **listw** est une liste créée par la fonction **nb2listw**, **n** est le nombre de “zones” considérées (le nombre de comtés dans notre exemple plus haut) et **S0** correspond à la somme globale des poids.

Les arguments **n** et **S0** peuvent sembler étranges puisque cette information est contenue dans l'objet **listw**, ils restent pour autant des arguments obligatoires.

Il nous faut tout de même modifier la façon dont l'on va générer notre matrice de poids puisque la fonction de **moran** prend en entrée une liste et non une matrice. Il nous suffit pour cela d'ajouter un argument à la fonction **nb2listw** pour générer des poids de distances binaires qui vérifient si les zones sont adjacentes et qui seront retournées dans une liste :

```
> ohio_lung.bw <- nb2listw(ohio_lung.nb, style = 'B')
```

On peut ainsi calculer le coefficient I de Moran de cette façon :

```
> moran(ohio_lung$LM68, ohio_lung.bw, n=length(ohio_lung.bw$neighbours), S0=Szero(ohio_lung.bw))
$I
[1] 0.05905108

$K
[1] 31.44218
```

La valeur du coefficient I peut varier de -1 à 1, comme un coefficient de corrélation plus classique. La valeur que l'on obtient ici signifie donc l'absence d'autocorrélation spatiale, l'on retrouvait en effet deux comtés qui présentaient beaucoup plus de cas que tous les autres et venait influencer fortement les comtés voisins une fois que l'on calculait la moyenne spatiale mobile, mais à tort.

Là est toute l'importance du coefficient de Moran qui offre un test formel pour vérifier la présence d'autocorrélation spatiale.

L'on peut en effet effectuer un test d'hypothèses à l'aide de la fonction `moran.test` qui prend comme paramètres :

```
> moran.test(x, listw)
```

Où `x` est un vecteur numérique, `listw` est une **matrice** créée par la fonction `nb2listw`.

```
> moran.test(ohio_lung$LM68, ohio_lung.bw)
```

```
Moran I test under randomisation
```

```
data: ohio_lung$LM68
```

```
weights: ohio_lung.bw
```

```
Moran I statistic standard deviate = 1.3672, p-value = 0.08578
```

```
alternative hypothesis: greater
```

```
sample estimates:
```

Moran I statistic	Expectation	Variance
0.059051077	-0.011494253	0.002662407

L'on retrouve donc la même valeur pour le coefficient I de Moran mais l'on obtient également une p-value qui est ici inférieure au seuil de 5%. On ne peut donc pas rejeter notre hypothèse nulle qui implique que les valeurs sont le résultat d'une distribution spatiale complètement aléatoire.

Une meilleure pratique serait même d'utiliser une simulation Monte-Carlo afin d'effectuer ces tests. Les valeurs que l'on passe à la fonction seront assignées de façon aléatoire aux polygones et le coefficient I de Moran sera calculé. Cette procédure sera répétée n-fois afin d'établir une distribution des valeurs attendues. La valeur observée du coefficient est ensuite comparée à la distribution simulée afin de déterminer à quel point il est probable que ces valeurs soient aléatoires.

Ce test est implémenté avec la fonction `moran.mc` qui prend comme paramètres :

```
> moran.mc(x, listw, nsim=99)
```

Où x est un vecteur numérique, $listw$ est une matrice créée par la fonction `nb2listw` et $nsim$ correspond au nombre de simulations, plus une, qui seront effectuées (on précise donc 99 ici afin d'en obtenir 100).

```
> moran.mc(ohio_lung$LM68, ohio_lung.bw, nsim=99)
Monte-Carlo simulation of Moran I

data: ohio_lung$LM68
weights: ohio_lung.bw
number of simulations + 1: 100

statistic = 0.059051, observed rank = 92, p-value = 0.08
alternative hypothesis: greater
```

L'on obtient pour autant ici des résultats relativement semblables à ceux obtenus précédemment, l'on rejette toujours notre hypothèse nulle.

5.3 Données géospatiales

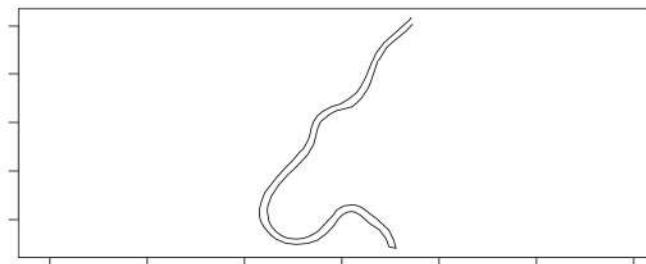
Pour cette partie du tutoriel, nous allons explorer l'interpolation des données géostatistiques continues à l'aide d'échantillons. Nous allons tenter de prédire et visualiser la concentration d'un polluant, le plomb, le long de la rivière Meuse aux Pays-Bas.

Deux packages seront nécessaires, le package `*sp*` qui facilite le stockage et la manipulation des données géostatistiques et le package `*gstat*` avec ses méthodes d'interpolation selon la méthode de krigeage.

```
> if(!require(gridExtra)) install.packages("gridExtra", repos = "http://cran.us.r-project.org")
> if(!require(sp)) install.packages("sp", repos = "http://cran.us.r-project.org")
> if(!require(gstat)) install.packages("gstat", repos = "http://cran.us.r-project.org")
> library(sp)
> library(gridExtra) # pour représenter plusieurs spplot ensemble
> library(gstat)
> data(meuse) # dans gstat
```

La région d'intérêt pour étude de pollution est le coude de la rivière Meuse. `Meuse.riv` est un objet matriciel que nous affichons avec la fonction de base de R, `plot`.

```
> data(meuse.riv)
> meuse.riv <- meuse.riv[which(meuse.riv[,2] < 334200 & meuse.riv[,2] > 329400),]
> plot(meuse.riv, type = "l", asp = 1)
```



Avant d'explorer notre jeu de données, nous transformons le dataframe "meuse" en dataframe spatiale (spdf). La fonction *coordinates* de *sp* prend les colonnes x et y pour cette opération.

```
> class(meuse) # meuse est initialement un datafram
[1] "data.frame"

> coordinates(meuse)=~x+y #maintenant un spdf
> class(meuse) # meuse est maintenant un SpatialPointDataFrame et les colonnes X et Y sont dans un slot de
[1] "SpatialPointsDataFrame"
attr(,"package")
[1] "sp"
```

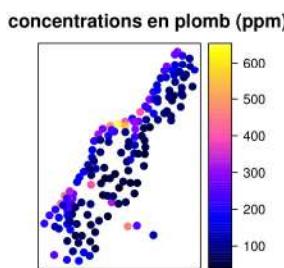
Voici les statistiques principales de notre variable d'intérêt le plomb (en ppm). Comme c'est une sortie d'un objet *sp*, *summary* sort des paramètres de base sur l'objet spatial dont les valeurs min et maximum des coordonnées x et y, le type de projection et le nombre d'observations.

```
> summary(meuse[c("lead")])

Object of class SpatialPointsDataFrame
Coordinates:
min     max
x 178605 181390
y 329714 333611
Is projected: NA
proj4string : [NA]
Number of points: 155
Data attributes:
  lead
Min.   : 37.0
1st Qu.: 72.5
Median :123.0
Mean   :153.4
3rd Qu.:207.0
Max.   :654.0
```

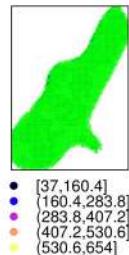
Pour visualiser, l'emplacement des 155 échantillons et leur concentration en plomb, nous faisons appel à *ssplot*, une fonction du package *sp*.

```
> spplot(meuse, "lead", colorkey = TRUE, main = "concentrations en plomb (ppm)")
```



Voici maintenant la surface sur laquelle nous allons faire l'interpolation (dataframe `meuse.grid`) que nous allons transformer également en objet spatial. Nous pouvons visualiser `meuse.grid` en vert à l'aide de `*spplot*`. Les points sont les sites d'échantillonnage. L'attribut `*sp.layout*` dans `spplot` permet de rajouter des couches au graphique.

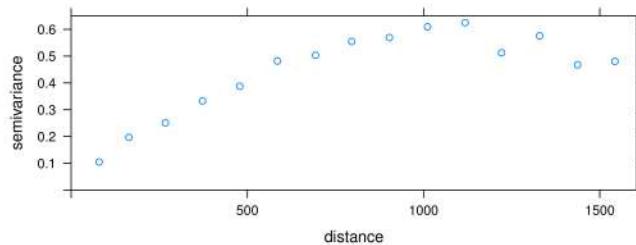
```
> data(meuse.grid)
> coordinates(meuse.grid)=~x+y
> spplot(meuse["lead"], sp.layout = list(meuse.grid,col='green'))
```



Après l'exploration et la visualisation des données, nous sommes à l'étape d'estimer la corrélation géographique de la concentration de plomb à l'aide du semi-variogramme.

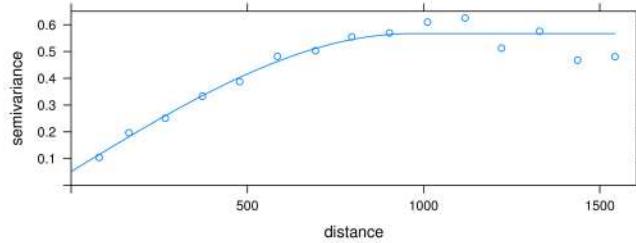
Le semi-variogramme décrit la différence de concentration entre deux points en fonction de leur distance respective. Il suffit d'utiliser la fonction `variogram` de `gstat`. Le paramètre `1` signifie que le type de krigage sera "ordinaire" (constant). Pour simplifier le modèle, nous supposons que la corrélation spatiale est isotropique (semblable dans toutes les directions). Dans le cas de données minières il est typique de prendre le log de la variable d'intérêt parce que le résultat est plus stable et la forme est semblable.

```
> vgm1 = variogram(log(lead)~1, meuse)
> plot(vgm1)
```



Maintenant, il s'agit de fitter les données du semi-variogramme avec un modèle (ie. estimer le modèle de corrélation). Par défaut `*fit.variogram*` estime les presque tous les paramètres. On ne doit choisir que le type de courbe ici sphérique ("Sph"). Il est possible d'ajuster le modèle manuellement. Il faut alors utiliser la courbe du semi-variogramme pour estimer les paramètres (de `*sill*`, `*range*` et `*nugget*`). Le `*sill*` étant la valeur de y à son palier, `*range*` est la valeur de x au début du palier et `*nugget*` l'ordonnée à l'origine.

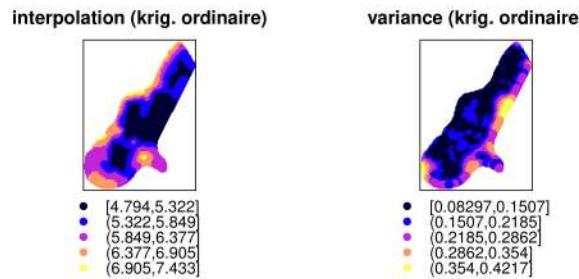
```
> vgm1.fit=fit.variogram(vgm1, vgm("Sph")) #fit.variogram modélise les données du variogramme expérimentale
> plot(vgm1, vgm1.fit)
```



La dernière étape est de faire l'interpolation des résultats (le krigage) et d'estimer la variance. `*grid.arrange*` permet d'afficher les deux résultats dans deux colonnes différentes.

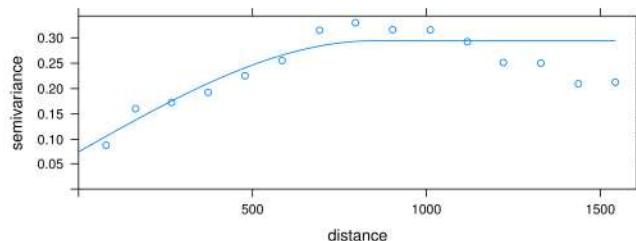
```
> vgm1.krige = krige(log(zinc)~1, meuse, meuse.grid, model = vgm1.fit)
[using ordinary kriging]

> grid.arrange (spplot(vgm1.krige["var1.pred"], main="interpolation (krig. ordinaire)"),
+                 spplot(vgm1.krige["var1.var"], main="variance (krig. ordinaire)", ncol = 2)
```

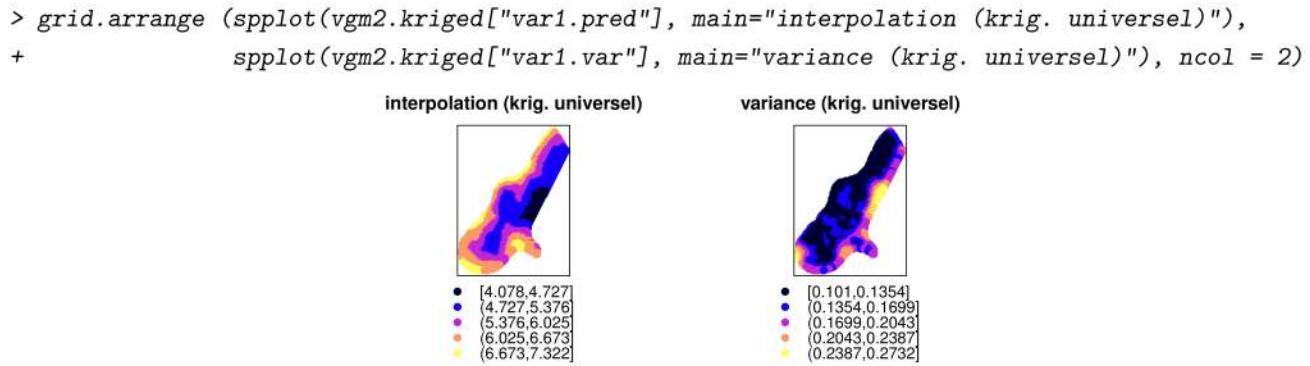


Si on suppose, que c'est la rivière qui charrie les polluants comme le plomb alors la distance entre la rivière et un emplacement de la carte est un facteur important. Dans ce cas, on veut faire du krigage "universel" (en fonction d'une autre variable). Il suffit donc de substituer le "1" du krigage "ordinaire" pour le facteur "dist" (distance entre la rivière et un point précis). On refait refit le variogramme, puis on refait l'interpolation pour avoir un résultat final.

```
> vgm2 = variogram(log(lead)~dist, meuse)
> vgm2.fit=fit.variogram(vgm2, vgm("Sph")) #fit.variogram modélise les données du variogramme expérimentale
> plot(vgm2, vgm2.fit)
```



```
> vgm2.krige = krige(log(zinc)~dist, meuse, meuse.grid, model = vgm2.fit)
[using universal kriging]
```



6 Bibliographie

- Adler, Jessie (s.d.) "intro-to-r/gis-with-R-intro.Rmd" . Récupéré de <https://github.com/jessesadler/intro-to-r/blob/master/gis-with-r-intro.Rmd>
- ArcGIS (s.d.) "Understanding geostatistical analysis". Récupéré de <https://desktop.arcgis.com/en/arcmap/latest/extensions/geostatistical-analyst/understanding-geostatistical-analysis.htm>
- Baillargeon, Sophie (2005) "Le krigage : revue de la théorie et application à l'interpolation spatiale de données de précipitations", Mémoire présenté à l'Université Laval. Récupéré de <https://www.mat.ulaval.ca/fileadmin/mat/documents/lrivest/EtudesGraduees/SBaillargeon.pdf>
- Bivand, Roger (2020-03-09) "CRAN Task View: Analysis of Spatial Data". Récupéré de <https://cran.r-project.org/web/views/Spatial.html>
- Bivand, Roger, Edzer Pebesma, Virgilio Gomez-Rubio (2005) "Applied Spatial Data Analysis With R ", Springer
- Choi, Mona (2013) "Book review : Spatial Analysis of Epidemiology". Récupéré de <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3717438/>
- Cressie, Noel 1993 "Statistics for Spatial Data, Revised Edition", Wiley & Sons
- Dale, Mark R. T., Marie-Josée Fortin (2014) "Spatial Analysis: A Guide For Ecologists by, 2nd edition" Cambridge University Press, p. 450
- De Smith, Michael, Michael Goodchild, Paul Longley (2018) "Geospatial Analysis - 6th edition, 2018", The Winchelsea Press.
- De Smith, Michael, Michael Goodchild, Paul Longley (2020) "Geospatial Analysis 6th Edition, 2020 update". Récupéré de <https://www.spatialanalysisonline.com/HTML/index.html>

- Elsvier (2020) "Journal of Spatial Statistic Author Information Pack 20 Mar 2020". Récupéré de <https://www.elsevier.com/journals/spatial-statistics/2211-6753?generatepdf=true>
- ESRI (s.d.) "History of GIS" , Récupéré de <https://www.esri.com/en-us/what-is-gis/history-of-gis>
- Gabriel, Edith (2010) "Introduction à la statistique spatiale", 42èmes Journées de Statistique, Marseille, France. Récupéré de <https://hal.inria.fr/inria-00494770/document>
- Gelfand, Alan, Peter Diggle, Peter Guttorp, Fuentes Montserrat (2010) "Handbook of Spatial Statistics", CRC Press, 619 p.
- Getis, A. \$(s.d.)\$ "Spatial Statistics". Récupéré de https://www.geos.ed.ac.uk/~gisteac/gis_book_abridged/files/ch16.pdf
- Gimond, Manuel (2019) "Chapter 11 Point Pattern Analysis" dans "Intro to GIS and Spatial Analysis". Récupéré de <https://mgimond.github.io/Spatial/point-pattern-analysis.html>
- GISGeography (s.d.) "Kriging Interpolation The Prediction Is Strong in this One" Récupé de <https://gisgeography.com/kriging-interpolation-prediction/>
- Karpatne, Anuj at al (2017) "Machine Learning for the Geosciences: Challenges and Opportunities". Récupéré de <https://arxiv.org/pdf/1711.04708.pdf>
- Lovelace, Robin, Jakub Nowosad, Jannes Muenchow (2020) "Geocomputation with R" . Récupér de [://geocompr.robinlovelace.net/](http://geocompr.robinlovelace.net/)
- Maxwell, Aaron (s.d.) "Semivariogram Explained". Récupéré de <https://www.youtube.com/watch?v=L-hnxGq74q0>
- Moat, Justin (2015) "Book review : Spatial Analysis: A Guide For Ecologists by, 2nd edition", Linnean Society Botanical Journal, vol. 179, issue 3, p. 550.
- Pfeiffer D, Robinson T, Stevenson M, Stevens K, Rogers D. Clements (2008) "Spatial analysis in epidemiology". Oxford, Oxford University Press.
- Smith, T.E., (2020) "Overview of Areal Data Analysis" dans "Notebook on Spatial Data Analysis". Récupéré de https://www.seas.upenn.edu/~ese502/NOTEBOOK/Part_III/1_Overview_of_Areal_Data_Analysis.pdf
- Smith, T.E., (2020) Notebook on Spatial Data Analysis. Récupéré de <http://www.seas.upenn.edu/~ese502/#notebook> <https://www.seas.upenn.edu/~ese502/#notebook>

- Souris, Marc, Laurence Bichaud (2011) " Statistical methods for bivariate spatial analysis in marked points. Examples in spatial epidemiology" Elsevier Journal of Spatial and Spatio-temporal Epidemiology, vol.2 p. 227-234.