Univerzita Palackého v Olomouci
Přírodovědecká fakulta
Katedra geoinformatiky

Tomáš BURIAN

ROZŠÍŘENÍ INTERPOLAČNÍCH NÁSTROJŮ V R PROJECT O MODELY NEJISTOTY

Bakalářská práce

Vedoucí práce: Mgr. Jan CAHA

Olomouc 2013

Čestné prohlášení

Prohlašuji, že jsem bakalářskou práci bakalářského studia oboru Geoinformatika a geografie vypracoval samostatně pod vedením Mgr. Jana Cahy.

Všechny použité materiály a zdroje jsou citovány s ohledem na vědeckou etiku, autorská práva a zákony na ochranu duševního vlastnictví.

Všechna poskytnutá i vytvořená digitální data nebudu bez souhlasu školy poskytovat.

V Olomouci 16. května 2013 Tomáš BURIAN

Děkuji vedoucímu práce Janu Cahovi za podněty a připomínky při vypracování práce.
Dále děkuji své rodině za poskytnuté prostředky a prostředí.

Obsah

ÚVOD

Problematika nejistoty je velmi často přehlížena. Lidé mnohdy ani netuší, že se s tímto fenoménem setkávají. Dokonce lze konstatovat, že nás obklopuje neustále, ať už v profesním či soukromém životě. Člověk má jistý odhad pro řadu popsatelných jevů, tento odhad navíc dokáže ještě upřesnit různými postupy či technologiemi. Avšak nakolik si můžeme být jisti, že je aplikovaná technika naprosto přesná? Existují ověřené teorie, které dokazují právě nepřesnost a tím i potvrzují existenci nejistoty výsledných dat. Nepřesné mohou být geodetická měření, kde se téměř vždy vyskytuje určitá odchylka od reality. Také metody sběru výškových dat, při snímání povrchu Země, vykazují zřejmou nejistotu v rámci až několika metrů.

Je tedy očividné, že nejistota opravdu existuje. Prof. Mikhail Kanevski v roce 2012 prohlásil, že je tato oblast nedostatečně prozkoumána a právě proto by bylo velice příhodné využít vyspělosti dnešního světa a aplikovat nejistotu všude tam, kde to jen lze. Samotné modely nejistoty jsou poté jednou z možností, jak toto tvrzení podpořit například v oblasti modelování zemského povrchu.

Spojením modelů nejistot a interpolačních nástrojů lze prakticky ukázat, jaké rozdíly ve výsledku způsobují variabilní vstupní data. Proto je nutné uvědomit si, že ne vždy je získaná hodnota ta správná a že může být ovlivněna nejistotou.

1 CÍLE PRÁCE

Cílem bakalářské práce je především rozšíření interpolačních nástrojů v softwaru R project o vybrané modely nejistoty. Dále rešerše používaných balíčků v R project a výběr nejvhodnějších z nich pro praktickou část práce. Mimo to v teoretické části práce stručně nastínit základy využití nejistoty pro modelování povrchů a základní využívané modely nejistoty.

V praktické části bude provedena implementace vybraných modelů nejistoty v R project formou funkcí, které vhodně rozšíří a doplní existující nástroje. Výstupem této části práce bude soubor funkcí kompatibilních s verzí R 3.0.0, které budou zabaleny ve výsledném R balíčku. Obsahem budou funkce pro tvorbu dat, gridu, modelů nejistoty a interpolace. Navíc bude připojena řádná dokumentace ve formě stručného návodu, jak tyto funkce používat. Celý balíček pak bude možno svobodně použít a případní uživatelé si budou moci prakticky vyzkoušet princip této práce. Textová část bude zpracována podle zásad dle šablony dostupné na webových stránkách katedry v typografickém softwaru Tex. Bude zde vysvětlena problematika nejistoty a rešerše balíčku pro R. Na závěr bude připojeno jednostránkové resumé v anglickém jazyce.

Celá bakalářské práce bude dokončena a odevzdána v daném termínu, a to i v digitální podobě na CD, a bude o ni vytvořena webová stránka v souladu s pravidly dostupnými na stránkách katedry.

2 POUŽITÉ METODY A POSTUPY ZPRACOVÁNÍ

2.1 Použitá data

Vzhledem k faktu, že vyhovující data pro práci nejsou příliš dostupná a pro ověření správné funkčnosti algoritmů jich nebylo zapotřebí, byla zvolena alternativní cesta k jejich dosažení. Tuto cestu zajišťuje několik funkcí, které byly v rámci praktické práce vytvořeny a ve výsledku byly schopny data samy generovat.

Z výše zmíněných důvodů byla veškerá použitá data v rámci bakalářské práce uměle vygenerována a byla nastavena tak, aby plně vyhovovala příkladnému chodu celého balíčku.

2.2 Použité programy

2.2.1 The R Project for Statistical Computing

Program The R Project for Statistical Computing (dále jen R) je volně šiřitelný software, dostupný na oficiálním webu r-project (Hornik, 2013). Díky licenci umožňující volné šíření se stal velmi populárním programem, který již zastínil mnoho komerčně založených řešení v rámci tematiky. Stejné tvrzení platí i o počtu uživatelů, kde se R ujalo prvenství, a počtem svých uživatelů předběhlo již dříve založené placené konkurenty. V posledních letech nestoupá jen jeho oblíbenost, ale i možnosti nabízející uživatelům. Rozšiřující se programátorská základna uživatelů má také za následek mnoho užitečných nástrojů, balíčků i funkcí (Burns, 2006).

Obrázek 1: Grafické logo softwaru (převzato z: R Core Team (2013))

Software patří do množiny softwaru GNU projektu. Tento projekt vznikl v roce 1984 a založil tím systém GNU, což je operační systém unixového typu. Základní myšlenkou byl vznik svobodného softwaru, tedy volně šiřitelného trendu. Celý program tvoří několik komponent a jádro. Mezi komponenty patří aplikace, knihovny a nástroje vývojářů, dále systémem pro komunikaci a cílení zdrojů směrem k hardwaru, což se označuje jako jádro (Free Software Foundation, 2013).

V neposlední řadě je vhodné zmínit i kvalitní uživatelskou podporu v České republice. Zde působí společnost RZJ - STAT s.r.o. se svou širokou společností odborníků. Hlavními účely jsou statistické poradenství, kurzy, software a programování. Společnost se mimo jiné podílela i na řadě balíčků, které jsou volně dostupné (Komárek, 2008 - 2012).

Stejně jako Tinn-R, R Commander, Red-R či několik dalších programů, tak i RStudio využívá programovacího jazyka R. Jedná se tedy o grafické uživatelské prostředí (interface) založené na jazyce R a prostředí, které umožňuje spravovat data a následně je podrobovat statistickým analýzám včetně možnosti grafického výstupu. Software nabízí široké statistické či testovací možnosti a díky implementaci programovacího jazyka R lze jednoduše, v případě potřeby, tyto možnosti rozšiřovat. A co více, vzhledem k vzájemné kompatibilitě, je možné využít i komerční programovací jazyk S, na jehož základě byl navržen jazyk R. Další silnou stránkou programu jsou také možné grafické výstupy, které kvalitně zastupují grafy umožňující zobrazení i takových prvků, jakými mohou být například matematické vzorce či symboly (Hornik, 2013; R Core Team, 2013).

Interface, tedy grafické rozhraní RStudia se sestává z 5 základních polí:

1. hlavní menu,
2. zdrojový kód,
3. konzole,
4. pole historie a pracovního prostředí,
5. pole pro načítání adresářů, grafů, balíčků a help.

Jako u víceméně každého programu je tu hlavní menu a panel nástrojů. Dále zdrojový kód, kde se zobrazují veškeré použité příkazy během tvorby projektu. Do tohoto pole je také možné rychle vkládat příkazy pro otevřený projekt. V poli konzole se zobrazuje aktuální průběh a procesy probíhající při práci, je zde relativně přístupné prostředí pro tvorbu cílených procesů. Historie zajišťuje přesný výčet veškerých dříve použitých příkazů. Velmi pozitivní záměr tvůrce programu je propojení jednotlivých polí, díky čemuž je umožněno aktivně i zpětně používat již dříve proběhnuté procesy, které se uložily do historie. Posledním objektem je pole pro načítání dat, dohledávání a aktivaci balíčků, zobrazování grafů a dokonce i online help pro podporu uživatele.

2.2.2 RTools 3.0

Druhý použitý software je nadstavbou pro R (viz výše). Zajišťuje tvorbu a správu balíčků pro R pod systémem Microsoft Windows. O první verzi se zasloužil Prof.

Brian Ripley, od té doby prošel software velkým vývojem a dnes je dostupný ve verzi 3. Samotný uživatel pocítí po instalaci především jednu změnu, a to právě uvnitř RStudia v podobě nové karty Build, která se v základní verzi nevyskytuje. Zde je poté možné balíčky přehledně spravovat (R Core Team, 2013).

2.3 Postup zpracování

Na počátku celé práce bylo nutné nejprve nastudovat patřičnou literaturu, a to především v oblasti problematiky nejistoty. Dalším důležitým bodem při studování teorie k bakalářské práci byla i oblast programování v R a studium žádaných balíčků pro R project. Na základě získaných znalostí byly vytvářeny rešerše vysvětlující základní poznatky a teorie v rámci jednotlivých sekcí.

Po získání potřebných informací a teoretického základu bylo přistoupeno ke tvorbě praktické části. Úspěšné a vyhovující podobě algoritmů a klíčových funkcí předcházelo samozřejmě sekundární studium zdrojových kódů a principů již vytvořených funkcí. Veškeré nabyté dojmy, proč použít právě R, pak potvrzuje myšlenka (Matloff, 2011): je krásné a levné, proč používat něco jiného? Přistoupilo se tedy k fázi programování nejprve funkcí pro generování dat a gridu. Vstupní data dostala základní atributy x, y, z a požadovaný typ třídy objektu *uncertainSpatialPoints*. Grid byl vytvořen na základě zvolených parametrů požadovaného rozměru. Pro úplnost procesu byl na počátku naprogramován i první model nejistoty, který se aplikoval na primární data a modifikoval je tak pro vstup do interpolačních funkcí. Posléze byly takticky naprogramovány interpolační funkce, na kterých se testovala funkčnost právě vygenerovaných dat a zároveň i správnost interpolačních procesů. Celkem byly zvoleny 3 druhy interpolací, a to metoda IDW, spline a kriging. Jako nástavba byla sestavena i funkce pro odhad variogramu, jenž vstupuje do kriging interpolace. Tato dodatečná procedura je určena spíše pro náročnější uživatele, kteří mají zájem o hlubší pohled na algoritmizaci krigingu. Do každé funkce vstupovala vygenerovaná data a po dokončení průběhu byl výsledkem nový objekt, se kterým lze dále pracovat.

Po vyřešení problematiky vstupu dat a interpolací byly vytvořeny modely nejistot, které lze aplikovat na jakákoli vstupní data požadovaného formátu. Určení nejistoty probíhá podle definovaných možností, jako například procentuální přesnost dat, náhodné hodnoty v intervalu či s využitím náhodné odchylky. V závěru praktické části byly všechny tyto funkce zabaleny do výsledného balíčku, který je možné volně připojit do softwaru RStudio. Balíček bude mít celkem 6 složek, v nichž se bude ukrývat několik užitečných funkcí pro tvorbu:

- dat,

- modelů nejistoty,
- požadované třídy objektů,
- gridu,
- interpolace (IDW, Spline, Kriging)
- variogramu.

3 BALÍČKY V R

Základem celého softwaru je jádro, které propojuje ostatní komponenty a zajišťuje stabilitu. Právě k tomuto jádru lze připojovat balíčky a tím zvyšovat funkcionalitu a rozšiřovat program o nové vlastnosti. V oficiální verzi R je obsaženo několik málo balíčků, které zajišťují primární operace. Seznam všech dostupných (známých) balíčků pro R project je veřejně dostupný na oficiální webové stránce *CRAN*. Zde je snadné dohledat, podle názvu či data vydání, všechny základní informace o balíčcích. Vypátrat je možné například aktuální verzi, autora, datum vydání či základní informace o balíčku (R Core Team, 2013).

R je velice mocný nástroj a dokáže pracovat s nespočetným množstvím procesů, od nejjednodušších elementárních výpočtů až po skutečně složité operace. To vše je dostupné v rámci prostředí R, a to díky možnosti rozšíření programu pomocí balíčků. Těchto balíčků pak existuje celá řada. V současnosti se jejich počet odhaduje na čtyři až pět tisíc a každým dnem se tento počet zvyšuje. Níže vypsané informace byly recenzovány dne 25. 1. 2013.

spatstat

Je jedním z nejznámějších balíčků. Obsahuje přes 1000 funkcí pro vykreslování prostorových dat, simulace, sestavování modelů. Provádí simulace, prostorové analýzy rozptylu bodů, testy, sestavování modelů a mnohé další. Dále analýzu prostorových dat, zejména prostorových bodových rastrů. Autoři jsou Adrian Baddeley and Rolf Turner a kol. Aktuální verze je 1.31-0 (Baddeley a Turner, 2005).

Maptools

Slouží pro čtení a manipulaci s prostorovými objekty. Obsahuje nástroje pro manipulaci a čtení geodat. Dále rozhraní pro spolupráci prostorových objektů s balíky jako například spatstat, maps, tmap, PBSmaping. Autoři jsou Nicholas J. Lewin-Koh and Roger Bivand a kol. Aktuální verze je 0.8-22 (R Core Team, 2013).

gstat

Je určen pro prostorové a časoprostorové geostatistické modelování, predikce a simulace. Přináší nástroje pro modelování variogramu, simple (co)kriging, ordinary (co)kriging, universal (co)kriging. Navíc při spojení s balíčkem automap poskytuje velké možnosti. Autorem je Edzer Pebesma a další. Aktuální verze je 1.0-15. Shodou

okolností byla z balíčku aplikována i stejnojmenná funkce *gstat*, a to pro interpolaci metodou IDW (Pebesma, 2004).

rgdal

Ve zkratce poskytuje vazby pro geoprostorové data. Zajišťuje plynulý chod veškerých operací v R. Dále umožňuje přístup k projekčním a transformačním operacím. Autoři jsou Timothy H. Keitt, Roger Bivand, Edzer Pebesma, Barry Rowlingson. Aktuální verze je 0.8-4 (R Core Team, 2013).

lattice

V prostředí R má na starosti především lattice grafiku. Jedná se o výkonný a elegantní vizualizační systém dat, s důrazem na data vícerozměrná. Autorem je Deepayan Sarkar. Aktuální verze je 0.20-13 (Sarkar, 2008).

geoR

Poskytuje analýzy nad geostatickými daty. Přesněji analýzy geostatické zahrnující tradiční, likelihood a Bayesian metody. Autory jsou Paulo J. Ribeiro Jr a Peter J. Diggle. Aktuální verze je 1.7-4. Z tohoto balíčku byla použita funkce *grf* pro pohodlné generování dat (Diggle a Jr, 2007; Jr a Diggle, 2001).

Hmisc

Název je zkratkou pro Harrell Miscellaneous. Poskytuje funkce pro analýzu dat, výpočty velikosti vzorku, import datasetů či pokročilé grafiky. Autoři jsou Frank E Harrell Jr a kol. Aktuální verze je 3.10-1 (R Core Team, 2013).

RSAGA

Balíček pro výpočty a analýzy terénu funkcemi programu SAGA GIS uvnitř softwaru R. Obsahuje také funkce pro správu ASCII formátu. Autorem je Alexander Brenning a aktuální verze je 0.93-1 (Brenning, 2008).

automap

Doslovně balíček určený pro automatické interpolace. V praxi funguje ve 2 krocích, nejprve automaticky odhadne variogram a poté zavolá balíček gstat pro dokončení procesu. Autor je Paul Hiemstra. Aktuální verze je 1.0-12. Z balíčku byly vybrány dvě funkce, pro výpočet kriging interpolace *autoKrige* a pro odhad variogramu *autofitVariogram* (Hiemstra et al., 2008).

intamap

Další varianta balíčku určeného pro automatické interpolace. Autory jsou Edzer Pebesma, Jon Olav Skoien a kol. Aktuální verze je 1.3-15 (Pebesma et al., 2010).

FuzzyNumbers

Balíček s nástroji a metodami pro práci s fuzzy čísly. V budoucnosti jej lze vhodně využít pro další praktický rozvoj celé bakalářské práce. Umožňuje například generování náhodných fuzzy čísel či aritmetické operace s fuzzy hodnotami. Autorem je Marek Gagolewski. Aktuální verze je 0.02 (Gagolewski, 2012).

fields

Ve zkratce lze tvrdit, že fields přináší nástroje pro prostorová data. Přesněji to pak jsou nástroje s důrazem na metody spline, prostorových dat a statistik. Autoři jsou Reinhard Furrer, Douglas Nychka and Stephen Sain. Aktuální verze je 6.7.6. Tento balíček poskytl pro bakalářskou práci funkci *Tps* pro spline interpolace (R Core Team, 2013).

RandomFields

Poskytuje Gaussian simulace, hodnoty náhodných polí čí podmíněné simulace. Autory jsou Martin Schlather, Peter Menck a kol. Aktuální verze je 2.0.66. Pro vytvoření korelované chyby byla z tohoto zdroje využita funkce *GaussRF* (R Core Team, 2013).

R.utils

Tento balíček je určen pro programátory a vývojáře R balíčků. Za zmínku stojí především jeho funkce *sourceTo*, která načte a vykoná Rkový kód v daném souboru.

Autor je Henrik Bengtsson. Aktuální verze je 1.23.2. Pro potřeby bakalářské práce byla použita především (již zmíněná) funkce *sourceTo* (R Core Team, 2013).

stats

Je jedním ze základních balíčků celého R projektu. Obsahuje nástroje pro statistické výpočty a generování náhodných hodnot. Na vývoji se nepodílel jen R Core Team, ale i řada přispěvatelů z celého světa. Dnes je dostupný ve verzi 3.0.0. Celý balíček je velmi užitečný, což bylo ověřeno i během tvorby funkcí v rámci bakalářské práce. Byly použity rovnou tři funkce, a to *predict* na tvorbu předpovědí u interpolačních procesů, dále funkce *rnorm* pro generování náhodné odchylky a poslední z nich byla *runif* pro výběr náhodné hodnoty z intervalu (R Core Team, 2013).

3.1 Shrnutí faktů

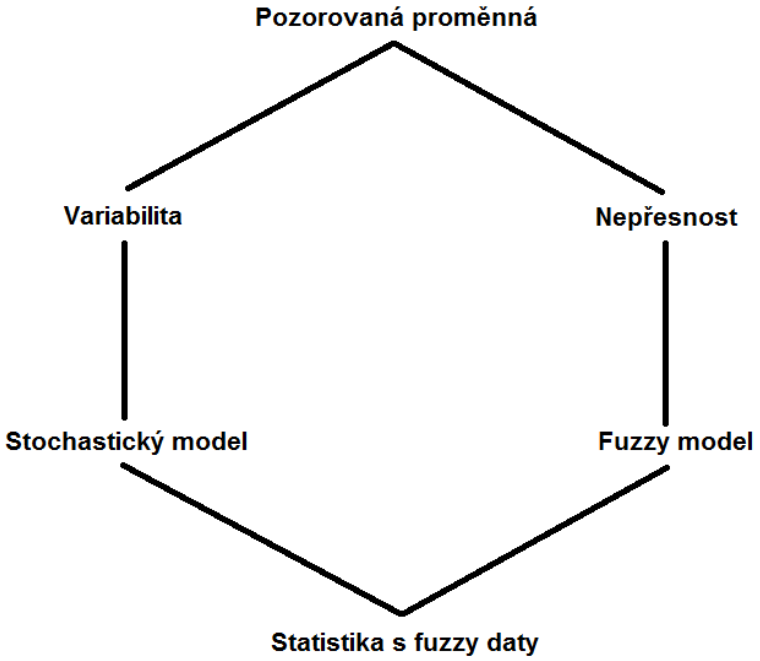
V praxi pak funguje celý systém balíčků jako jedna velká baterie funkcí a procesů, které lze jednoduše dohledat a použít. Nejjednodušší cesta k získání balíčku s požadovanými funkcemi vede přes help konzoli v RStudiu, kde jej můžeme vyhledávat na základě klíčových slov. Případně je přípustné využít i online databázi balíčků CRAN či internetový vyhledávač. Jednotlivé balíčky na sebe mohou vzájemně navazovat a kombinovat se. Toto tvrzení lze prokázat na příkladném balíčku automap, který pro vykonání procesu interpolace metodou kriging, vyžaduje funkce z gstat balíčku. Takto vznikají originální soubory i stovek funkcí uvnitř několika různých balíčků, které jsou určitým způsobem spjaty dohromady a doplňují se.

4 MODELY NEJISTOTY

Nejistota se sama o sobě může vyskytovat prakticky kdekoli. Jedná se o přirozenou součást reálného světa a posléze i námi vytvořeného obrazu libovolného charakteru (Caha, 2013). Otázkou pak zůstává, jak k ní dochází, co je příčinou a jak se tohoto faktu vyvarovat. Podle řady publikací od různých autorů je pravdou, že za touto problematikou stojí několik klíčových faktorů. Veškerá digitální data v sobě nesou určité chyby i nejistotu, které je možné snadno odhalit srovnáváním více datových sad pořízených různými metodami v rámci jednotného území (Fisher a Tate, 2006). Na geografické scéně vědy se s takovými daty setkáváme poměrně často a ačkoli je tato nejistota prvotně zmíněna a jistým způsobem vnímána, nedostává takový prostor, jaký by si zasloužila (Caha, 2013).

4.1 Složky nejistoty

Byly definovány dvě základní složky nejistoty, a to nepřesnost a variabilita (Viertl, 2011). Přičemž na počátku všeho stojí určitá pozorovaná proměnná. Měření této proměnné probíhá vždy se stálou či nestálou přesností, která vytváří právě nepřesnost v datech a tím zakládá na významu nejistoty. Variabilita pak také může, díky nestálým faktorům, při opakovaném měření vytvářet nejisté podmínky pro další práci. Tyto dvě složky jsou tedy základní kameny pro pochopení samotné nejistoty.

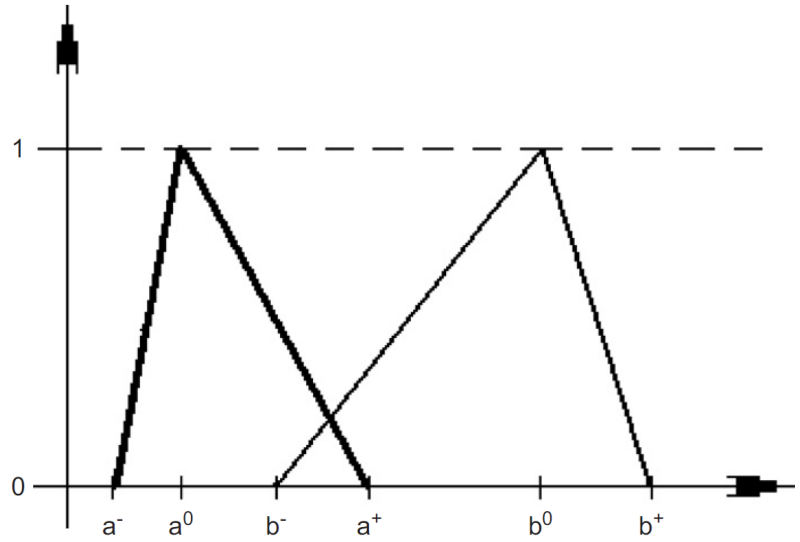


Obrázek 2: Složky nejistoty (upraveno dle: Viertl (2011))

Při logickém využití nepřesnosti a variability lze problematiku zkoumat i hlouběji (Caha et al., 2013). Je možné nad nimi vytvářet stochastické i fuzzy modely a z těchto modelů je pak možné konstatovat, že můžeme pracovat se statistikou na fuzzy datech (viz obr. 2). V rámci bakalářské práce byla brána v potaz cesta tvorby nejistoty přes nepřesnost a fuzzy modely. Jedná se tedy o nejistotu založenou na přesnosti jednotlivých měření, kde je očekávána určitá nepřesnost. Naproti tomu stojí využití variability dat, kdy je měření bráno jako absolutně přesné a zkoumají se rozdíly mezi několikrát opakovanými měřeními. Avšak tato možnost v rámci práce šetřena nebyla.

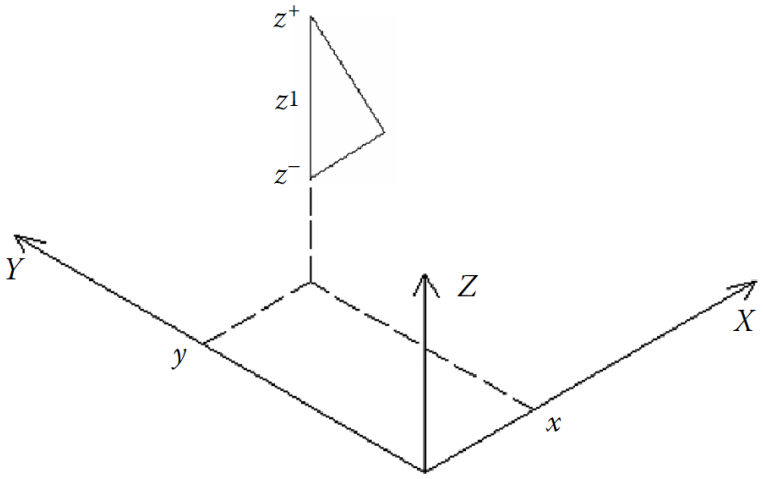
4.2 Fuzzy data

Z matematického hlediska může být většina provedených měření uznána jako fuzzy hodnoty a to proto, že není možné zajistit absolutní přesnosti naměřených výsledků. Navazující myšlenkou je pak tvrzení, že interpolační data nelze považovat za reálné hodnoty, nýbrž za určité rozsahy těchto hodnot. Vše bývá způsobeno v důsledku chyb a nejistoty. Tyto poznatky lze poté využít právě v modelování nejistoty (Waelder, 2007).



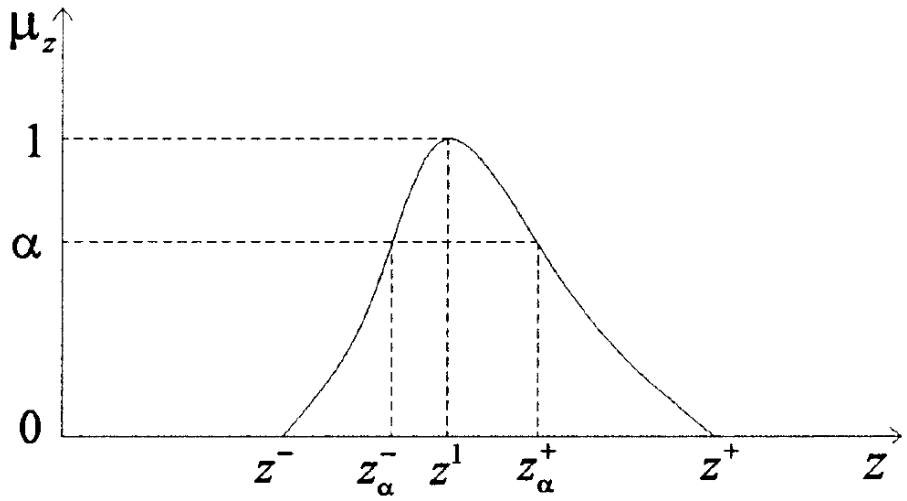
Obrázek 3: Fuzzy intervaly proměnné (převzato z: Waelder (2007))

Jednotlivé naměřené hodnoty pak mohou být reprezentovány jako fuzzy intervaly (viz obr. 3), což lze i označit za nejjednodušší fuzzy objekty. Interval $A = [a^-, a^0, a^+]$ je přesně hraničně vymezen a obsahově definován zkoumanou proměnnou a^0 a jejími limitujícími (dolními a^- a horními a^+) hodnotami (Waelder, 2007). Takto upravené hodnoty pak mohou reprezentovat proměnné i v prostoru (viz obr. 4). V našem případě je tímto způsobem modifikovaná hodnota proměnné na souřadnici z, která nabývá již definovaného fuzzy intervalu $Z = [z^-, z^1, z^+]$ (Santos, 2008).



Obrázek 4: Fuzzy reprezentace prostorové proměnné (převzato z: Santos (2008))

Modelování zemského povrchu je velmi známou oblastí GIS. Existují všeobecně platné výpočty a postupy, které se touto oblastí zabývají. Ovšem, jak efektivní jsou právě tyto teorie při zohlednění nejistoty dat? Jedno z možných řešení (Santos et al., 2002) pojednává o vyjádření nejistoty s využitím právě fuzzy čísel. Klíč této metody pro získání prostorové nejistoty, spočívá v (již výše zmíněném) porovnávání výškových dat s daty z přesnějších zdrojů. Tímto způsobem se odvozuje střední kvadratická chyba, která bývá běžně využívána na mnoha světových pracovištích, například USGS. Nejistota pak bývá vyjádřena převedením výškových hodnot dat na fuzzy čísla. Implementací právě fuzzy hodnot, reprezentující zkoumaný element dat, se zabývá i tato práce.



Obrázek 5: Převedená výšková data na fuzzy hodnoty (převzato z: Santos et al. (2002))

4.2.1 Interpolace

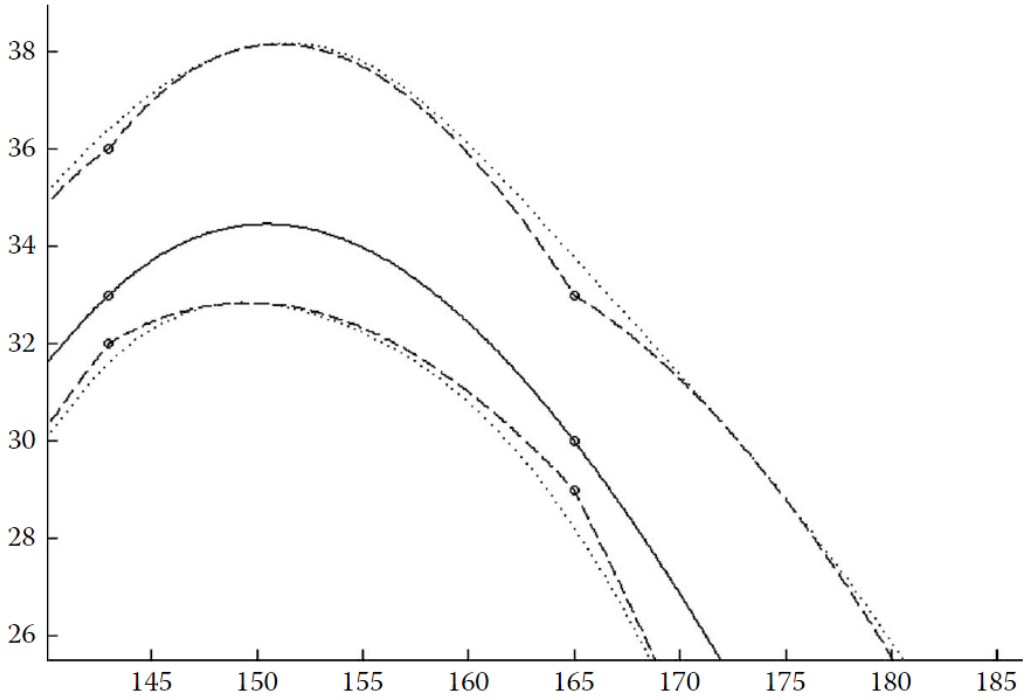
K zajištění interpolace je nutné definovat dva základní prvky ke vstupu do celého procesu. Těmito prvky jsou primární data a grid. Samotný grid jako zvolená mřížka pro interpolované hodnoty určuje pouze výsledné uspořádání dat. Na druhé straně data, po implementaci modelu nejistoty, mohou výsledek značně modifikovat.

Mějme tedy představu (Lodwick et al., 2008), že získaná výšková data z jsou reprezentována fuzzy intervaly $\widetilde{z}_i = [z_i^-, z_i^0, z_i^+], i = 1, ..., n$ a každá naměřená hodnota odpovídá právě jedné souřadnicové pozici $(x_i, y_i), i = 1, ..., n$. Všechna měření jsou také nepravidelně rozmístěna, a tudíž nesouhlasí s daty ve zvoleném gridu (Waelder, 2007). Jestliže mají být takto definovaná data interpolována ke gridu $\{X_j, Y_k\}, j = 1, ..., N, k = 1, ..., M$, pak výsledné interpolované hodnoty výšky $\widetilde{Z}_{jk} = [Z_{jk}^-, Z_{jk}^0, Z_{jk}^+]$ budou také fuzzy čísly a jejich vlastnosti budou záviset na zvolené interpolační metodě (Santos, 2008).

Když na takto upravená data aplikujeme metodu interpolace IDW, dostaneme následující výsledek (Waelder, 2007):

$$\begin{aligned}
&\widetilde{Z}_{jk} = \alpha_1^{jk}\widetilde{z}_1 + \alpha_2^{jk}\widetilde{z}_2 + ... + \alpha_n^{jk}\widetilde{z}_n, \\
&\sum_{i=1}^{n} \alpha_i^{jk} = 1, \alpha_i^{jk} = f(d_i^{ij}), d_i^{jk} = (x_i - X_j)^2 + (y_i - Y_k)^2, \\
&\qquad i = 1, ..., n, j = 1, ..., N, k = 1, ..., M. \qquad (1)
\end{aligned}$$

Pro vyšší srozumitelnost tvrzení lze také uvést grafické znázornění, kde je tato fuzzy myšlenka zachycena. Jako příklad, pro větší obsáhlost zvolených interpolačních metod, byl zvolen fuzzy prostorový spline (viz obr. 6). Na obrázku jsou patrné na první pohled hned tři neznámé. První z nich jsou druhy linií, kde plná představuje průběh interpolace středních hodnot, čárkované pak hranice fuzzy intervalů a tečkované linie vyznačují ideální stopu podle střední hodnoty. Druhou markantní záležitost reprezentují vyznačené trojice bodů, které vstupují do interpolací. Postupně směrem k vyšším hodnotám na svislé ose vymezují proměnné spodní hranice, střední hodnoty a horní hranice fuzzy intervalů. Třetí, tedy poslední a nejdůležitější, je pak průběh samotné interpolace. Zde je možné vidět jisté odlišnosti jednotlivých interpolací mezi body. Zatímco interpolace středních hodnot je plynulá, tak v hraničních situacích se její průběh poněkud mění a vybočuje od ideální stopy výsledné interpolace (Santos, 2008).

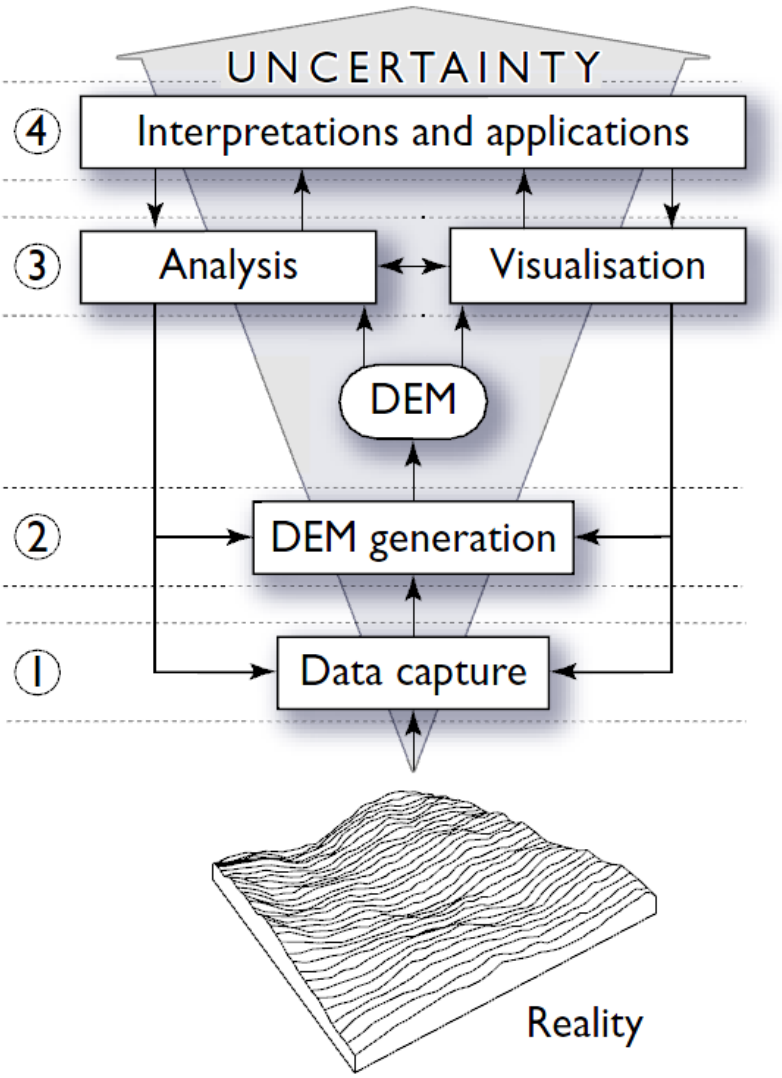


Obrázek 6: Výsledek fuzzy prostorového splinu (převzato z: Santos (2008))

Veškeré procedury, vytvořené v rámci této bakalářské práce, pak pracují na podobných (výše uvedených) principech. Existují tedy určitá základní data, která mají být podrobena jistým analýzám. Tato data jsou však před samotným postupem obohacena o nejistotu a až poté vstupují jednotlivě do dalších fází výzkumu. Detailní popis těchto procedur je vysvětlen v následující kapitole.

4.3 Nejistota v modelech povrchu

V oblasti tvorby digitálního výškového modelu (angl. digital elevation model, DEM) Země lze snadno demonstrovat příkladný vznik chyb. Což ve výsledku dává vzniknout nejistotě, jejíž míra může postupně gradovat už v průběhu celého pracovního procesu (viz obr. 7). Na počátku veškerých prací je samozřejmě sběr dat a již zde se právě vyskytuje prvotní zárodek nejistoty, který bývá dále jen umocňován. Přispívají k tomu postupně jak metody pro generování a tvorbu modelu, tak i další použité druhy analýz a vizualizací. Na konci stojí logicky interpretace a aplikace získaného výsledku, jenže i zde se může vyskytnout řada chyb, které mohou podtrhnout výslednou nejistotu celého procesu tvorby (Oksanen, 2006).

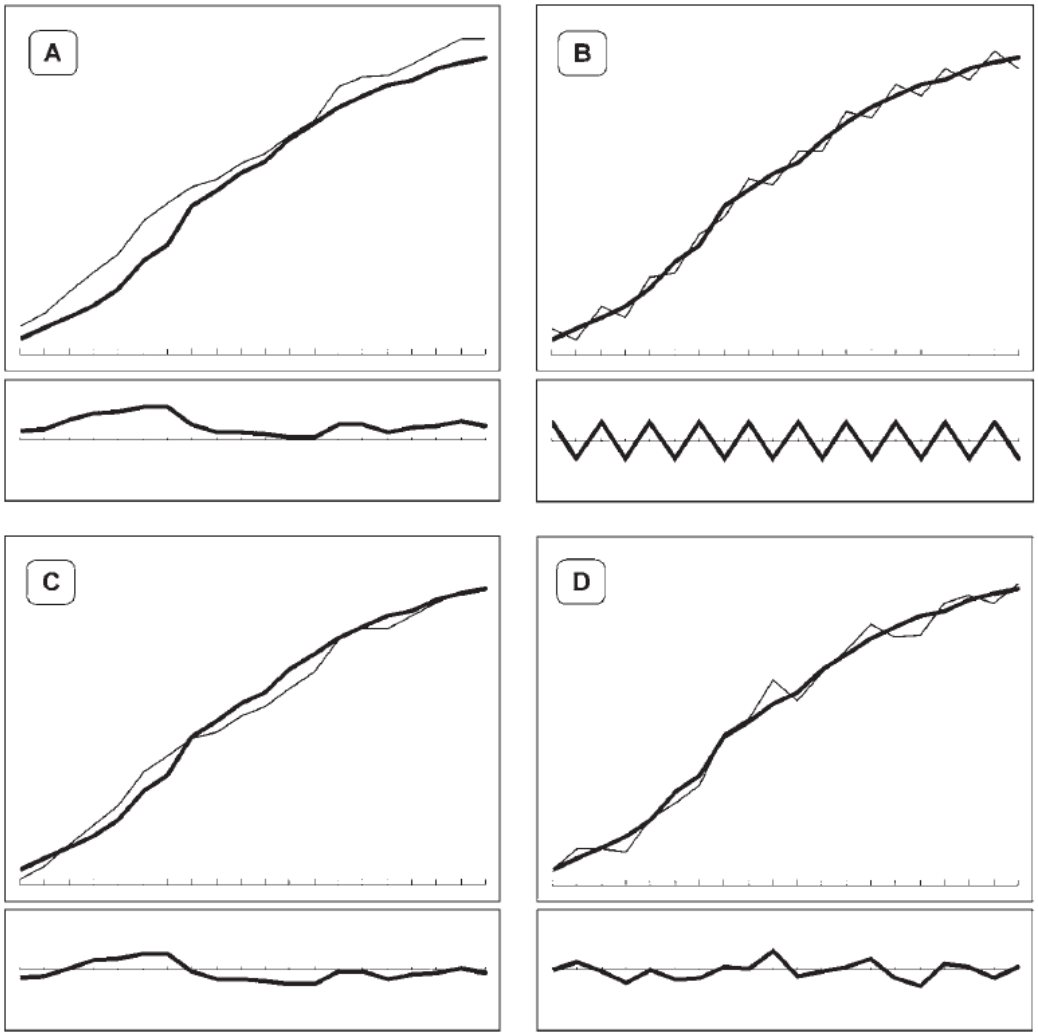


Obrázek 7: Tvorba DEM a představení nejistoty na základě zvoleného postupu práce (převzato z: Oksanen (2006))

4.3.1 Typy chyb

Chyby v DEM se mohou vyskytovat v horizontálním i vertikálním směru dat. Většinou však bývají tyto nepřesnosti uváděny jen u vertikálních, tedy výškových záznamů. Původ chyb lze zařadit do tří charakteristických kategorií (Fisher a Tate, 2006):

1. hrubé (angl. gross errors/blunders),
2. systematické (angl. systematic errors),
3. náhodné (angl. random errors).

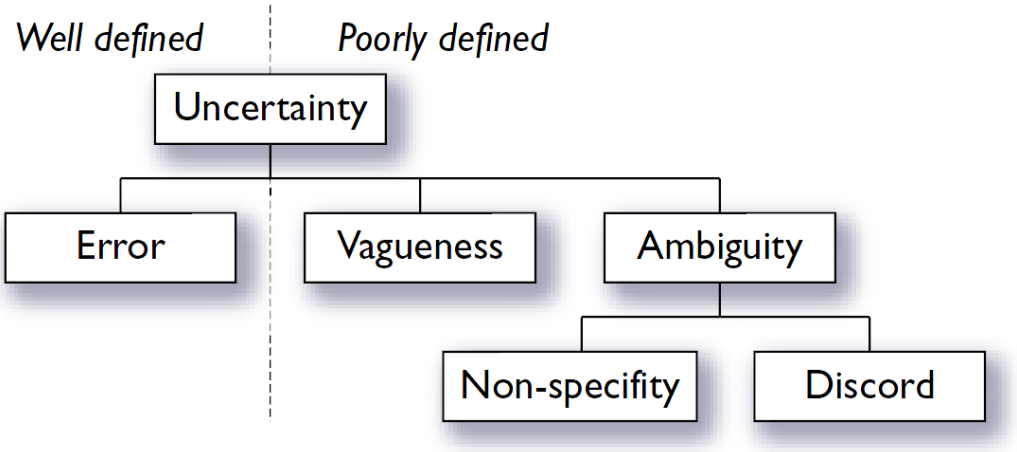


Obrázek 8: Porovnání profilu DEM a výskytu chyb. Výskyt: (A, B) systematických chyb, (C) prostorově autokorelovaných chyb (normální stav), (D) náhodné chyby (bez prostorové autokorelace). V každé situaci znázorňuje horní obrázek povrch země silnou čarou a povrch s chybami čarou tenkou, spodní obrázek zobrazuje samotnou chybu. (převzato z: Fisher a Tate (2006))

První zmíněná kategorie hrubých chyb může mít svůj původ například v selhání lidského faktoru či použité aparatury. Avšak hrubé chyby lze poměrně snadno rozpoznat i odstranit. Dalšími chybami jsou chyby systematické (viz obr. 7 - A, B), které se vyskytují podle určitého pravidla. Příkladem systematických chyb můžou být tzv. duchové, nebo-li obrysy linií identifikovatelné v mnoha DEM odvozených od vrstevnicových dat, nebo také známé umělé terasy. Poslední kategorie náhodných chyb (viz obr. 7 - C, D) pochází z vysoké rozmanitosti měření, nelze je zcela vhodně modelovat a bývají koncepčně reprezentovány náhodnými variacemi okolo střední hodnoty (Fisher a Tate, 2006).

4.3.2 Vlivy chyb

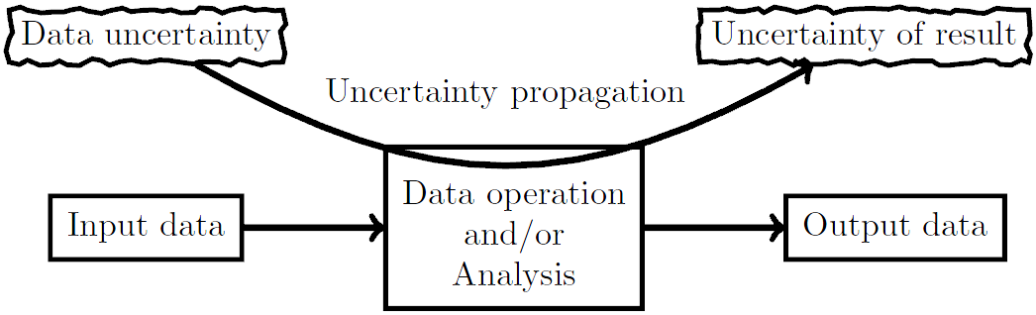
Vliv samotných chyb na nejistotu pak lze rozčlenit podle správnosti jejich definování (viz obr. 9). Toto členění má dvě základní skupiny, kam lze chyby zařadit, a to chyby správně stanovené a chyby stanovené nedostatečně (Oksanen, 2006). Na jedné straně stojí nejistota jako celek a její složkou jsou chyby (nebo také odchylky). Což je právě případem dobře definovaných chyb, protože mají statistickou povahu, nicméně v praxi jsou těžko dosažitelné z důvodu absence vhodných referenčních dat. Na straně druhé figuruje nejistota s podsložkami vágnosti a dvojznačnosti. Obzvláště pak vágnost je důležitá, jelikož její interpretací se zabývá tato bakalářská práce.



Obrázek 9: Definice chyb a jejich vliv na nejistotu (převzato z: Oksanen (2006))

4.4 Význam nejistoty

V současné době je běžné vytvářet povrchové modely nad získanými daty bez zdlouhavého vyšetřování správnosti dat, či dokonce ověřování nejistoty v poskytnutých datech. Lidé tak oficiálně zanedbávají, ať už vědomě či nevědomě, potřeby žádoucí kvality a přesnosti výsledků. Důvody, proč tomu tak je, jsou prosté. Pracovníci zkrátka nemají potřebný čas, technologie, finance či znalosti k těmto úkonům (Heuvelink, 2002). Přitom existuje hned několik principů, které byly publikovány a mohly by být využitelné i v praxi. Důležitá je přitom myšlenka (Caha et al., 2013), kdy nejistota na vstupu dat může, více či méně, ovlivnit i samotný výsledek provedené práce nad daty (viz obr. 10).



Obrázek 10: Role nejistoty (převzato z: Caha et al. (2013)

Jestliže tento fakt (viz obr. 10) přijmeme a uznáme tím i výskyt nejistoty v datech, pak bychom tomuto jevu měli věnovat minimálně zvýšenou pozornost. Již počáteční neošetřená nejistota (na vstupu) se totiž dále v průběhu procesu může rozrůstat. Protože v hraničních případech, kdy jsou vstupní data skutečně nepřesná, může snadno docházet k velmi rozdílným a nepřesným výsledkům (Caha et al., 2013).

4.5 Vlastní pojetí nejistoty

Sestavení vlastních modelů nejistoty nad vstupními daty pro účely bakalářské práce probíhalo podle výše uvedených myšlenek. Stěžejní inspiraci tvořily především publikace od autorů Santos, Viertl a Lodwick. Nejistota tudíž byla určována podle očekávané nepřesnosti při sběru vstupních dat a vytvořené modely nejistoty pak byly postaveny na possibilistickém vyjádření dat, nikoli na statistice.

Byla tedy využita teorie o reálném vzniku nejistoty na základě nepřesností při sběru primárních dat. Tato nepřesnost poté dala vzniknout právě jednotlivým typům modelů pro vytváření nejistoty, které modifikovaly vstupní data.

5 BALÍČEK UNCERTAINTYINTERPOLATION

Název balíčku je složen ze slov uncertainty a interpolation, což v českém překladu znamená nejistota a interpolace. Oficiální zkrácený název pro R pak zní UncerIn, pod touto zkratkou bude figurovat v prostředí R. Balíček byl vytvořen pro praktické vyzkoušení funkčnosti bakalářské práce. Hlavním cílem bylo ověření platnosti, zda je možné propojení a zprovoznění právě modelů nejistoty a interpolačních procesů. Tato hypotéza byla potvrzena a přijata. V rámci práce tedy nebylo plánováno zabývat se parametrizací, či jinými detailnějšími principy zvolených funkcí v rámci interpolačních procesů. Z tohoto důvodu je zde poměrně velký prostor pro případné rozšiřování celého projektu (viz kapitola 7).

Obsahem celého balíčku jsou kompletní nástroje pro tvorbu dat. Dále prostředky k tvorbě modelů nejistoty či generování gridů. Funkce pro tři druhy interpolačních procesů. A navíc jsou zde přiloženy funkce pro manipulaci s daty a odhad variogramu.

5.1 Třídy dat

Interpolace v R vyžadují jistý vytyčený vstup dat, samotný proces výpočtů je také pevně daný. Je zde nutný vstup několika parametrů, bez kterých není možné dosáhnout správného výsledku. Těmito vstupy jsou např. souřadnice, grid, interpolované proměnné či další vybrané parametry funkcí. Vzhledem k jisté míře automatizace všech funkcí a nastudovaným faktům, bylo přistoupeno k vytvoření vlastního typu dat o známém formátu a v objektovém návrhu verze *S3*. V současnosti existují dva tipy objektů, a to *S3* a *S4*, přičemž první z nich je vývojově starším a dnes stále dominujícím tipem v prostředí R (Matloff, 2011). Navíc *S3* je výrazně jednodušší pro programování a i z tohoto důvodu byl zvolen právě tento typ objektů (Hadley, 2013). Celý objekt pak dostává známou, později vždy na vstupu testovanou charakteristickou třídu, která zajišťuje správnost dat.

Programový kód 1: Zabalení a určení třídy dat (z funkce spatialPoints)

```
colums = c("x", "y", "z")
data = list(x = data[, colums[1]], y = data[, colums[2]], z = data
   [, colums[3]])
class(data) <- {"spatialPoints"}
```

Výše je uvedena část kódu pro převod objektu do třídy *spatialPoints*, k čemuž byla vytvořena stejnojmenná funkce spatialPoints. Název této funkce v sobě skrývá pojem prostorové body, čímž naznačuje, že v dalších krocích budeme pokračovat v prostoru

a je důležité, aby to bylo zřejmé i ze samotné třídy objektu. Tento kód zabaluje data pomocí příkazu *list* do podoby souřadnic *x, y, z* a přidělí jim vybraný typ třídy. Vzhledem k faktu, že pro navázání dalších kroků je vyžadována třída prvků *spatialPoints*, tak je zde uvedena na prvním místě a je preferována. Avšak nelze opomenout i další důležité možnosti, které mohou být při práci žádány. Z tohoto důvodu je, díky dalším přiloženým funkcím, možné vytvářet i třídy typu *dataframe* či *spatialPointsDataFrame*. V tomto kroku je tedy dostatečně pojištěna správnost vstupu dat s využitím očekávané třídy dat. Nespornou výhodou navíc je, že tuto funkci lze aplikovat na jakákoli data o správném formátu, tedy matici o sloupcích *x, y, z*. Uživatelé si tak mohou své data libovolně převádět do správné formy pro další procesy.

Data dostala tři primární atributy *x, y, z*, kde první dva reprezentují souřadnice a třetí je zkoumaná proměnná. Zároveň jsou generované hodnoty uspořádávány do požadovaných sloupců.

Avšak *spatialPoints* nebyla jedinou třídou, která byla vytvořena. Po implementaci modelů nejistoty data dostávají dva nové sloupce, tím rozšiřují objekt, a ten poté již nevyhovuje dosavadní třídě. Z tohoto důvodu byla přidána ještě druhá funkce, která převádí objekt do nové třídy *uncertainSpatialPoints*. Na vstupu jsou zde očekávány dva možné formáty dat. Prvním formátem je matice, kde je očekávána jistá struktura sloupců. Pakliže by sloupce nebyly vyhovující, tak bude proces ukončen a uživateli se zobrazí chybové hlášení. V druhém případě, kdy na vstupu funkce nebude matice, ale jakýkoli jiný formát, je řešení mnohem flexibilnější. Stačí na vstupu funkce definovat jednotlivé sloupce, které musí být stejné délky. Tím je podchycen a vyřešen vstup různých formátů dat, což řada uživatelů jistě ocení.

Programový kód 2: Formáty vstupních dat (z funkce uncertainSpatialPoints)

```
uncertainSpatialPoints.matrix <- function (data)
uncertainSpatialPoints.default <- function (x = NULL, y = NULL,
   uncertaintyLower = NULL, modalValue = NULL, uncertaintyUpper =
    NULL)
```

Podobně jako spatialPoints i tato funkce převádí data do žádaných formátů s tím rozdílem, že upravuje názvy atributů a kontroluje jejich existenci. Atributy, po propojení kódu modelu a této konverze dat, pak nabývají názvů pro souřadnice *x, y* a jednotlivé hodnoty nejistoty jsou pojmenovány jako *uncertaintyLower, modalValue* a *uncertaintyUpper*. Anglické názvy výsledků nejistoty představují hodnoty dolní hranice, střední hodnoty a horní hranice. Výsledkem jsou poté opět data ve vyhovujícím formátu, který splňuje veškeré předpoklady pro další využití.

5.2 Generování testovacích dat

Tvorba dat byla založena na myšlenkách potřeb pro získání tří primárních atributů bodů. Tyto atributy pak představují souřadnice *x, y* a hodnota zkoumané proměnné *z*. Vytvořené hodnoty poté byly využity jako testovací data výsledného balíčku.

Programový kód 3: Tvorba primárních dat (z funkce generateRandomPoints)

```
simPoints <- grf(numberOfPoints, grid = "irreg", cov.pars = c(
    sill, range), nug = nugget, cov.model = covModel, aniso.pars =
     c(anisotropyDirection, anisotropyRatio), xlims = xlim, ylims
    = ylim)
geoPoints = cbind(simPoints$coords, simPoints$data)
data = geoPoints
colnames(data) <-  c("x", "y", "z")
```

Uvedená část kódu generuje právě primární data, kde lze na počátku spatřit definici funkce *grf* z balíčku geoR. Tato funkce generuje náhodné gausovské pole bodů dle zadaných kovariačních parametrů. Základním parametrem je zde na prvním místě požadovaný počet generovaných bodů a poté řada parametrů specifikujících výsledné hodnoty (limity osy x a y, sill, nugget, range, kovarianční model a nastavení anisotropie), které lze libovolně nastavit podle potřeby. Následuje výběr dat pomocí příkazu *cbind* nejprve do formy matice. Ve vytvořené matici data dostávají klíčové názvy sloupců, tedy x, y, z, které jsou v pozdější fázi procesu očekávané a tvoří tak základní stavební kámen již zmíněné správnosti dat.

5.3 Modely nejistoty

Nyní, když máme základní data, můžeme přistoupit ke tvorbě modelů nejistoty. Zjednodušeně lze tvrdit, že se jedná o modifikaci zkoumané proměnné uvnitř dat (základního atributu *z*). Pro demonstraci tvorby modelů nejistoty v R lze uvést například model zakládající se na procentuální nepřesnosti. Představme si vzorovou situaci, kdy nejmenovaná společnost získá kontrakt ke zpracování dat o určitém území. Zkreslení výsledných dat, vlivem různých chyb při sběru, se může pohybovat řekněme na hranici 3 procent oproti skutečnosti. Podobná myšlenka byla i inspirací pro tento model nejistoty. Čili na vstupu funkce jsou vstupní data (naměřeně hodnoty) a zmíněná hodnota procentuální nepřesnosti dat (počet procent). Nejprve byla určena chybová odchylka od původních dat. Poté byla tato množina hodnot převedena do pomocné proměnné *modify*. Výsledky pak musely být ještě přepočítány do absolutních hodnot, čímž se předešlo případné chybě v dalších výpočetních procesech. Touto chybou je myšlena kolize při matematickém odčítání hodnot, kde může nastat nežádaná situace

odečítání záporného čísla, jelikož z procesu odčítání se stane sčítání. Pro vytvoření samotného modelu nejistoty pak již jen stačilo, v případě určení dolní hranice, odečíst tuto proměnnou od zkoumané množiny vstupních dat. V opačném případě, tedy určení horní hranice, byly ke vstupním datům tyto hodnoty přičteny. Na konci funkce dostaneme zpět vstupní data a k nim přidané hraniční nepřesnosti v rámci zadaného rozpětí. Stejným postupem pak byly vytvořeny i ostatní funkce pro vytváření nejistoty. Tyto vybrané metody byly založeny na modifikaci zkoumané proměnné pomocí konstantní hodnoty, korelované chyby, náhodné procentuální či numerické nepřesnosti v rámci intervalu a také za pomoci náhodné odchylky.

Programový kód 4: Výpočet nejistoty na základě procentuální nepřesnosti (z funkce uncertaintyPercent)

```
percent = (data$z/100) * numberOfPercent
modify = abs(percent)
uncertaintyLower = data$z - modify
uncertaintyUpper = data$z + modify
```

V rámci bakalářské práce bylo vytvořeno celkem 6 funkcí, tedy 6 různých modelů, které nejistotu na vstupních datech vytváří. Všechny názvy kódů byly navrženy tak, aby vždy začínaly pojmem uncertainty (nejistota) a pokračovaly vystihujícím výrazem, který charakterizoval onu výslednou nejistotu. Jak je již z názvů patrné, jsou jednotlivé modely založeny na matematické kalkulaci. Z hlediska principu průběhu se od sebe příliš neliší. Jediným razantním rozdílem je samotná kalkulace výsledného modelu, která probíhá podle zvoleného typu nejistoty. Na vstupu funkcí jsou vždy data a parametry pro výpočet nejistoty. Data budou na počátku ihned zkontrolována, zda jsou vyhovující. Tato kontrola je však vždy dodržena díky předešlé editaci dat do očekávaného formátu. Posléze následuje průběh tvorby modelu nejistoty a na závěr jsou veškeré výsledky zabaleny a vráceny uživateli v objektu třídy *uncertainSpatialPoints*. Tento typ třídy je logicky dále očekáván na vstupu pro interpolace.

uncertaintyConstant

První model se zakládá na konstantní nepřesnosti dat. Jedná se zároveň o nejjednodušší postup, který byl vytvořen. Výsledný model nejistoty byl vypočítán pouhým odečítáním či přičítáním definované konstanty od zkoumané proměnné. Takový případ nepřesnosti může nastat například při leteckém snímání povrchu, kdy bývá zcela běžně přikládána informace o naměřených výškových datech, které se mohou od skutečných hodnot lišit i o několik metrů. Právě podobné situace může řešit tento model, který implementuje k získaným datům nejistotu pomocí konstanty reprezentující odlišnost vůči realitě.

Programový kód 5: Ukázka funkce pro tvorbu modelu nejistoty o konstantní nepřesnosti (z funkce uncertaintyConstant)

```
uncertaintyConstant <- function (data, constant)
  uncertaintyLower = data$z - constant
  uncertaintyUpper = data$z + constant
```

uncertaintyError

Tento případ kalkuluje výslednou nejistotu na základě prostorově korelované chyby. Proces zajišťuje funkce *GaussRF*, kde podle (na vstupu funkce) definovaných parametrů probíhá generování chyb, které jsou poté vstupem pro výpočet samotného modelu.

Programový kód 6: Část funkce určující model s prostorově korelovanou chybou (z funkce uncertaintyError)

```
  error <- GaussRF(x=data$x, y=data$y, model="stable", grid=
     FALSE,param=c(0, 0.1*sill, 1,range/3 , 1))
  error = abs(error)
  uncertaintyLower = data$z - error
  uncertaintyUpper = data$z + error
```

uncertaintyPercent

Další vytvořený model nejistoty je založen na nepřesnosti procentuální a byl popsán již v předcházejícím textu práce (viz výše). Zajímavostí je, že vznikající nejistota je počítána přímo ze vstupních dat, k čemuž je nutné zadat pouze procento nepřesnosti.

uncertaintyRandomPercent a uncertaintyRandomNumber

Tyto dva modely nejistoty jsou ve výčtu záměrně uvedeny společně, jelikož si jsou vývojově velmi podobné. Principiálně byla jejich tvorba založena na výběru náhodné hodnoty z definovatelného intervalu. Přičemž první z nich je veden relativní cestou (procentuální interval) a druhý cestou absolutní (číselný interval). Hraniční meze intervalu je možné pohodlně nastavit na vstupu funkce pomocí parametrů, čímž je zajištěna dostatečná flexibilita procesu pro různě velké rozsahy intervalů. Výběr náhodných intervalů pak vykoná funkce *runif*, které k pohodlnému průběhu postačí zadat pouze počet bodů a hranice žádaného intervalu.

Programový kód 7: Tvorba modelu pomocí náhodné procentuální hodnoty z intervalu 0 až 3 (z funkce uncertaintyRandomPercent)

```
uncertaintyRandomPercent <- function (data, min=0, max=3)
  randomPercent = runif(length(data$z), min, max)
  percent = (data$z/100) * randomPercent
  modify = abs(percent)
  uncertaintyLower = data$z - modify
  uncertaintyUpper = data$z + modify
```

uncertaintyRandomDeviate

Poslední vytvořený model je postaven na myšlence náhodné odchylky od zkoumané proměnné ze vstupních dat. Hodnoty této veličiny byly určeny pomocí funkce *rnorm*, které stačí parametricky zadat počet výsledných hodnot, střední hodnotu a hodnotu směrodatné odchylky.

Programový kód 8: Ukázka R kódu pro výpočet modelu na základě nádodné odchylky (z funkce uncertaintyRandomDeviate)

```
  value = rnorm(length(data$z), mean, sd)
  modify = abs(value)
  uncertaintyLower = data$z - modify
  uncertaintyUpper = data$z + modify
```

5.4 Grid

Samotný princip použitých interpolačních funkcí je pevně daný. Kromě vstupních dat a jejich parametrů vyžadují na vstupu i grid. Zvolený grid poté slouží jako mřížka pro výsledné hodnoty interpolace. Samozřejmostí tedy je, že se ve výsledném balíčku bude vyskytovat i patřičná funkce pro generování vlastních gridů uživatele. Díky této možnosti nebylo nutné kvůli náročnosti interpolačních procesů složitě vyhledávat další data či vyhovující funkce pro tvorbu gridů. A co více, jejich vytváření není vůbec složité. Byly sestrojeny dvě možné cesty, jak tuto mřížku vytvořit. První a zároveň jednodušší cestou je zavolat tuto funkci a na vstupu předložit objekt třídy *uncertainSpatialPoints* a požadované rozměry. Zmíněný druh třídy objektu byl pevně stanoven i popsán výše, a právě proto bylo velmi výhodné tohoto formátu dat využít pro vytažení vstupních souřadnic gridu. Rozměry mřížky jsou pak udávány počtem požadovaných buněk ve směru osy x a y. Druhá cesta by se mohla zdát o něco složitější. Nicméně, jediným rozdílem oproti první cestě, je zadávání souřadnic gridu. Zde musí

být zvoleny manuálně. Parametry rozměrů mřížky zůstaly však ve stejném formátu a jejich volba se nijak neliší od předchozí možnosti.

Programový kód 9: Parametry funkcí pro tvorbu gridů (z funkce prepareGrid)

```
function (uncertainSpatialPoints, numberOfCellsX, numberOfCellsY)
function (x, y, numberOfCellsX, numberOfCellsY)
```

5.5 Interpolace

Otázka interpolačních algoritmů byla vyřešena formou tří funkcí a jednou nástavbou. Tyto funkce reprezentují tři druhy interpolací, a to metody IDW, spline a kriging. Nástavbu pak představuje kód pro odhad variogramu, který vstupuje do procesu krigingu. Prakticky vrací uživateli automaticky vybraný model variogramu, včetně jeho definujících hodnot range, sill a nugget.

Protože úkolem práce nebylo zabývat se parametrizací nebo detailnějšími principy interpolačních funkcí, postačily na vstupu pouze 2 parametry pro úspěšný chod celé interpolační procedury. Těmi byla vstupní data a zvolený grid. Parametrizace interpolací byla převzata z přidružených originálních, tedy původních, interpolačních funkcí. Po spuštění procesu budou nejprve otestována vybraná vstupní data, zda jsou ve správném požadovaném formátu. Poté přicházejí na řadu úpravy dat do takové podoby, aby mohla být aplikována do interpolací.

5.5.1 Kriging

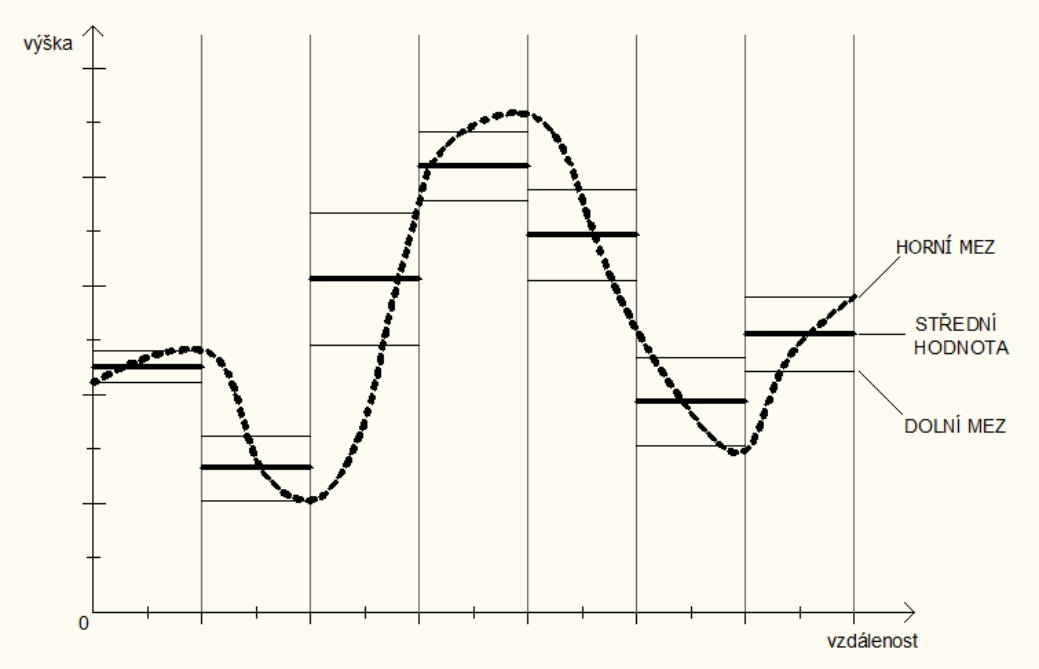
Interpolace metodou kriging byla zajištěna díky funkci *autoKrige*, která je součástí balíčku automap a vytváří automatické interpolace krigingu. Na začátku celého procesu nejprve proběhnou úpravy dat. Prvním krokem je zde převod vstupních objektů do tříd *dataframe*, což usnadňuje následnou manipulaci s daty. Především pak nastavení souřadnic pro data a grid. Zde byla využita výhoda definovaného formátu třídy *uncertainSpatialPoints*, a proto byly souřadnice označeny výběrem jednotlivých sloupců z dat. Po zvolení souřadnic vstupují, společně s gridem, jednotlivé zkoumané proměnné do interpolací.

Programový kód 10: Interpolační funkce (z funkce kriging)

```
z_L_krig = autoKrige(uncertaintyLower~x+y, data_frame, grid_frame)
z_krig = autoKrige(modalValue~x+y, data_frame, grid_frame)
z_U_krig = autoKrige(uncertaintyUpper~x+y, data_frame, grid_frame)
```

Zmíněné proměnné jsou samozřejmě střední hodnota a její dolní, horní hranice vypočítané na základě modelu nejistoty. Postupně tedy proběhnou interpolace nad

třemi proměnnými. Průběh procesu je poté možné graficky popsat (viz obr. 11) jako průchod interpolace všemi třemi proměnnými, interpolace je znázorněna přerušovanou linií, střední hodnoty proměnné jsou vyznačeny tučně a její hraniční meze pak čarami jednoduchými.



Obrázek 11: Průběh interpolace

Ze získaných výsledků byly posléze vybrány pouze ty partie, které obsahují vypočítané predikce pro jednotlivé body všech proměnných, tedy zájmové výsledky interpolací. Na závěr bylo vše opět zabaleno a převedeno, podobně jako u modelů nejistot, do formátu *uncertainSpatialPoints*. Případný uživatel tak získává kompletní objekt se souřadnicemi a výsledky interpolací ve všech bodech.

6 Instalace R balíčku

Vyprodukovaný R balíček UncerIn bude dostupný online, a to na webové adrese katedry geoinformatiky univerzity Palackého v Olomouci:

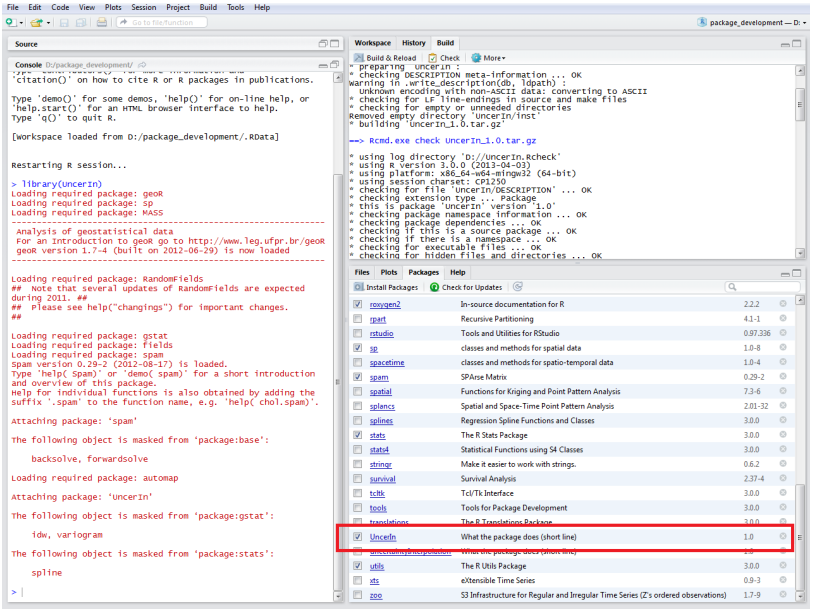
http://www.geoinformatics.upol.cz/dprace/bakalarske/burian13//.

Zde si jej může volně stáhnout a nainstalovat každý zájemce o tuto práci. Případně jsou uvolněna i veškerá práva pro svobodné šíření tohoto balíčku. Formát, v jakém bude balíček dostupný pro uživatele, bude jednoduchý zkomprimovaný a bude jej možno po stažení snadno použít či rozbalit. A jak vlastně balíček nainstalovat? První krok by měl vést na výše uvedenou webovou adresu, kde si balíček lze stáhnout a uložit. Dále spustit software R a zde vede nejjednodušší cesta přes příkaz *install.package()* pro lokalizaci a nahrání balíčku. Do závorky tohoto příkazu se uvádí několik parametrů pro definici procesu, avšak my si vystačíme pouze se dvěma:

1. *file.choose()*: definuje název vybraného souboru, v našem případě zůstávají závorky prázdné a to proto, aby byla uživateli nabídnuta možnost vyhledání správného balíčku,

2. *repos=NULL*: definuje cílové uložiště (např. CRAN), kde se žádaný balíček nachází, ponecháním hodnoty *NULL* se stává uložištěm pravě harddisk uživatele.

Výsledný příkaz pak může vypadat například takto *install.package(file.choose(), repos=NULL)*. Po vložení tohoto příkazu do softwaru R se uživateli otevře nové okno pro dohledání staženého balíčku, a to přímo na pevném disku počítače uživatele. Pak již jen stačí balíček najít, označit a nainstalovat.



Obrázek 12: Načtený balíček v prostředí R

7 Diskuze

Vzhledem k faktu, že podobný projekt v R doposud nebyl zpracován, je nutné zaujmout specifický postoj k této bakalářské práci. Především pak přijmout tvrzení, že primárním cílem bylo otestovat, zda je vůbec možné implementovat teorii nejistoty k interpolačním procesům v prostředí R. Nejednalo se tedy o snadný úkol. Nicméně díky mnohým námětům a nápadům, ze stran autora i vedoucího práce, byla celá práce dovedena k pozitivnímu výsledku.

V návaznosti na předchozí myšlenku je také nutné opět podotknout, že nebylo cílem zabývat se důkladnější parametrizací či zkoumáním detailnějších principů použitých R funkcí. Celá práce tedy ve výsledku nabízí vytvořený balíček poskytující základní procesy pro představení problematiky v prostředí R. Naskytuje se zde tak obrovský případný potenciál pro další návaznost a rozvoj práce. Rozšíření může být realizováno hned v několika směrech, ať už v detailnějším nastavení využitých funkcí, propojením projektu s dalšími balíčky obsahujícími stejný tematický podtext, či celkovým rozšiřováním flexibility balíčku o nové funkce. Například zmíněná parametrizace vstupních funkcí, kdy uživatel na počátku procesu dostane možnost volby detailnějšího nastavení celého průběhu, může sama o sobě prohloubit možnosti interpolačních procesů či zvýšit kvalitu výstupních dat. Celý balíček může být v budoucnu obohacen také o funkce provádějící vizualizace dat a tím poskytnout i výsledné užitečné grafické obrazy. V otázce propojení projektu a dalších balíčků, které souvisí s nejistotou či řešenou problematikou celkově, se zde nabízí kupříkladu balíček FuzzyNumbers s nástroji a metodami pro práci s fuzzy čísly.

V oblasti praktického využití lze výsledek práce aplikovat v několika sférách. Je přeci globálním trendem těžit ze situací pokud možno co nejlepší výsledky, a to bez razantně zvýšených finančních nákladů. Jako nejvýhodnější možnost se pak nabízí oblast GIS modelování zemského povrchu, kde by mohlo dojít ke zlepšení přesnosti výsledných modelů terénu. Další alternativy aplikace balíčku mohou být vhodně zvoleny na základě výskytu nejistoty, kde je efektivní implementovat model nejistoty a rozšířit tak vyjádření vstupních dat, prohloubit možnosti interpolačních procesů a také i komplexnost výstupních dat. V neposlední řadě je zde možnost využití i ve školském sektoru pro praktické ukázky problematiky.

8 Závěr

Práce se celkově zabývá praktickým rozšiřováním existujících interpolačních funkcí v softwaru R project o předpokládanou nejistotu ve vstupních datech. Na počátku bylo nutné nejprve nastudovat množství literatury zabývající se problematikou nejistoty. Ze získaných poznatků byla vytvořena rešerše na toto téma a společně s rešerší R balíčků tvořily základní tvůrčí myšlenky pro další práci. Dále bylo také zapotřebí dosáhnout dostatečné znalosti v oblasti programování a práce s R.

Hlavním cílem bakalářské práce pak především bylo otestovat možnosti R v interpolacích obohacených o nejistotu, a případně tyto interpolační nástroje rozšířit o modely nejistoty. Tento cíl byl patřičně prověřen a uznán jako realizovatelný. Proto bylo přistoupeno k programování a tvorbě funkcí zajišťující tyto procesy a ve výsledku byly zahrnuty do R balíčku, který tvoří stěžejní výstup práce. Celkem bylo vytvořeno 14 funkcí. Tyto funkce slouží pro tvorbu primárních dat, správných datových tříd, modelů nejistoty, generování gridu, interpolací i odhadu variogramu.

Cíle bakalářské práce byly tedy splněny, a to možná i nad počáteční očekávání. Důkazem tohoto tvrzení je právě vytvořený balíček pro R, který je plně funkční, a potvrzuje tím zmíněné hypotézy.

LITERATURA

BADDELEY, A., TURNER, R. Spatstat: an R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12, 6, s. 1–42, 2005. Dostupné z: `www.jstatsoft.org`. ISSN 1548-7660.

BRENNING, A. Statistical geocomputing combining r and saga: The example of landslide susceptibility analysis with generalized additive models. In *SAGA – Seconds Out (= Hamburger Beitraege zur Physischen Geographie und Landschaftsoekologie, vol. 19)*, s. 23–32. J. Boehner, T. Blaschke, L. Montanarella, 2008.

BURNS, P. R relative to statistical packages: Comment 1 on technical report number 1 (version 1.0) strategically using general purpose statistics packages: A look at stata, sas and spss. Technical report, Statistical Consulting Group UCLA Academic Technology Services, 2006. Dostupné z: `http://www.burns-stat.com/pages/Tutor/R_relative_statpack.pdf`.

CAHA, J. *PROPAGACE NEJISTOTY V ANALÝZÁCH FUZZY POVRCHŮ*. Teze disertační práce, Univerzita Palackého v Olomouci, 2013.

(v recenzním řízení) CAHA, J., VONDRÁKOVÁ, A., DVORSKÝ, J. Mathematical models of uncertainty for surface analyses and decision making. 2013.

DIGGLE, P. J., JR, P. J. R. *Model Based Geostatistics*. New York : Springer, 2007.

FISHER, P. F., TATE, N. J. Causes and consequences of error in digital elevation models. *Progress in Physical Geography*, 30, 4, s. 467–489, August 2006. ISSN 03091333.

FREE SOFTWARE FOUNDATION, I. Operační systém gnu, 2013. Dostupné z: `http://www.gnu.org`.

GAGOLEWSKI, M. *FuzzyNumbers Package: Tools to deal with fuzzy numbers in R*, 2012. Dostupné z: `http://www.ibspan.waw.pl/~gagolews/FuzzyNumbers/`.

HADLEY. S4 hadley/devtools wiki github, 2013. Dostupné z: `https://github.com/hadley/devtools/wiki/S4`.

HEUVELINK, G. B. M. Analysing Uncertainty Propagation in GIS: Why is it not that Simple? In FOODY, G. M., ATKINSON, P. M. (Ed.) *Uncertainty in remote sensing and GIS*, s. 307. Chichester : Wiley, 2002. ISBN 0470844086.

HIEMSTRA, P., PEBESMA, E., TWENHÖFEL, C., HEUVELINK, G. Real-time automatic interpolation of ambient gamma dose rates from the dutch radioactivity monitoring network. *Computers & Geosciences*, 2008. DOI: http://dx.doi.org/10.1016/j.cageo.2008.10.011.

HORNIK, K. The R FAQ, 2013. Dostupné z: `http://CRAN.R-project.org/doc/FAQ/R-FAQ.html`.

JR, P. J. R., DIGGLE, P. J. geoR: a package for geostatistical analysis. *R-NEWS*, 1, 2, s. 14–18, June 2001. Dostupné z: `http://CRAN.R-project.org/doc/Rnews/`. ISSN 1609-3631.

KOMÁREK, A. Rzj - stat s.r.o., 2008 - 2012. Dostupné z: `http://www.rzj-stat.cz/index.html`.

LODWICK, W., ANILE, M., SPINELLA, S. Introduction. In LODWICK, W. (Ed.) *Fuzzy surfaces in GIS and geographical analysis : theory, analytical methods, algorithms, and applications*, s. 1–46. Boca Raton : CRC Press, 2008. ISBN 9780849363955.

MATLOFF, N. *THE ART OF R PROGRAMMING A Tour of Statistical Software Design*. William Pollock, 2011.

OKSANEN, J. *Digital Elevation Model Error in Terrain Analysis*. PhD thesis, University of Helsinki, 2006.

PEBESMA, E., CORNFORD, D., DUBOIS, G., HEUVELINK, G., HRISTOPOULOS, D., PILZ, J., STOEHLKER, U., MORIN, G., SKOIEN, J. Intamap: the design and implementation of an interoperable automated interpolation web service. *Computers & Geosciences*, ?, s. ?, 2010. Dostupné z: `http://dx.doi.org/10.1016/j.cageo.2010.03.019`.

PEBESMA, E. J. Multivariable geostatistics in s: the gstat package. *Computers & Geosciences*, 30, s. 683–691, 2004.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. Dostupné z: `http://www.R-project.org`.

SANTOS, J. Surface modeling. In LODWICK, W. (Ed.) *Fuzzy surfaces in GIS and geographical analysis : theory, analytical methods, algorithms, and applications*, s. 85–104. CRC Press, 2008. ISBN 9780849363955.

SANTOS, J., LODWICK, W., NEUMAIER, A. A New Approach to Incorporate Uncertainty in Terrain Modeling. In EGENHOFER, M., MARK, D. (Ed.) *Geographic Information Science*, 2478 / *Lecture Notes in Computer Science*, s. 291–299. Springer Berlin Heidelberg, 2002. ISBN 978-3-540-44253-0.

SARKAR, D. *Lattice: Multivariate Data Visualization with R.* New York : Springer, 2008. Dostupné z: `http://lmdvr.r-forge.r-project.org`. ISBN 978-0-387-75968-5.

VIERTL, R. *Statistical methods for fuzzy data.* Chichester, West Sussex : Wiley, 2011. ISBN 9780470699454.

WAELDER, O. An application of the fuzzy theory in surface interpolation and surface deformation analysis. *Fuzzy Sets and Systems*, 158, 14, s. 1535–1545, July 2007.

SUMMARY

The work deals with the practical expansion of existing interpolation functions in software R project by using the uncertainty of input data. These extensions are represented in the ability to create several different models of uncertainty over the input data. The implementation will expand the possibilities in declarations of input data, improve the possibilities of interpolation processes, and also the complexity of the output data. The actual data uncertainty can be based on the expected vagueness of the variable. This vagueness can be putted into the form of spatially correlated errors, random values, percentage of values, constant or random deviations from the original data. Created uncertainty models are built on possibilistic expression of data, not on statistics. After that the data are going into the interpolations processes that creates model of the variable containing the uncertainty of the input. After the end of interpolation it is possible to keep on working and process the obtained results in the software R, what could be the possible way how to improve the results of this bachelor thesis. The result of the entire work is a package for R, that contents functions to provide the exemplary data creation, including adjustments to the stated format. Additionally 6 functions for creating the models of uncertainty were programmed. Three types of basic interpolations (spline, kriging, IDW) and a function to create grids for the interpolations. The main objective of this bachelor thesis was to test the options of R interpolation methods enriched by the uncertainty. This is therefore the first work of its kind that can be further developed and expanded. From the author's point of view, it was all about creating an imaginary first step, finding the actual informations and creating of the foundation stone in this so far under-explored area.

Seznam příloh

Příloha 1 - CD s daty, webovými stránkami a textem práce