

**Univerzita Karlova**  
**Přírodovědecká fakulta**

Katedra aplikované geoinformatiky a kartografie

Studijní program: Geografie (bakalářské studium)

Studijní obor: Fyzická geografie a geoinformatika

3. ročník



Anna SVÁTKOVÁ

**NÁVRH WEBOVÉ GISOVÉ APLIKACE PRO ANALÝZU  
LETOKRUHOVÝCH CHRONOLOGIÍ A PREZENTACI JEJÍCH  
VÝSLEDKŮ**

**DESIGN OF A WEB GIS APPLICATION FOR ANALYSIS OF TREE-RING  
CHRONOLOGY DATA AND PRESENTATION OF ITS RESULTS**

Bakalářská práce

Vedoucí práce: Mgr. Lukáš Brůha, Ph.D.

Praha, 2021

**Vysoká škola:** Univerzita Karlova v Praze  
**Katedra:** Aplikované geoinformatiky a kartografie

**Fakulta:** Přírodovědecká  
**Školní rok:** 2020/2021

# Zadání bakalářské práce

**pro** Annu Svátkovou  
**obor** Fyzická geografie a geoinformatika

**Název tématu:** Návrh webové GISové aplikace pro prezentaci a analýzu letokruhových chronologií

## Zásady pro vypracování

Bakalářská práce navrhne webovou GISovou aplikaci, která bude umožňovat prezentaci výsledků modelování predikce reakce hlavních lesních dřevin na změny geografických charakteristik. Za tímto účelem bude adaptován postup vytvoření chronologií extrémního snížení růstu (Cropperových hodnot) a jejich prostorové interpolace. Tento postup bude automatizován v podobě geoprocessingového nástroje, který bude možné publikovat jako webovou službu. Takový webový nástroj umožní uživateli webové aplikace provádět na základě zadaných parametrů vlastní analýzy. Při volbě technického řešení bude zohledněno, že výsledná webová aplikace bude administrována v rámci ArcGIS Enterprise technologie.

**Rozsah grafických prací:** dle potřeby

**Rozsah průvodní zprávy:** cca 40 - 60 stran

**Seznam odborné literatury:**

ArcGIS Web AppBuilder (Developer Edition), ESRI 2020. Accessed online: <https://developers.arcgis.com/web-appbuilder/guide/get-started.htm>

Brewer, P. W., Guiterman C. H. A new digital field data collection system for dendrochronology. *Dendrochronologia*. 2016;38:131–5

Brewer, P. W. TELLERVO, A guide for users and developers. 2016. Accessed online: <http://tellervo.org/support/tellervo-manual.pdf>

Nétek, R. *Webová kartografie - specifika tvorby interaktivních map na webu*. 2020. ISBN 9788024458274

Vedoucí diplomové práce: Lukáš Brůha, Ph.D.

Konzultant diplomové práce:

Datum zadání diplomové práce: 01.12.2020

Termín odevzdání diplomové práce: podzim 2021

*Platnost tohoto zadání je po dobu jednoho akademického roku.*

.....  
Vedoucí diplomové práce

.....  
Vedoucí katedry

V Praze dne: 1.12.2020

## **Prohlášení**

Prohlašuji, že jsem závěrečnou práci zpracovala samostatně a že jsem uvedla všechny použité informační zdroje a literaturu. Tato práce ani její podstatná část nebyla předložena k získání jiného nebo stejného akademického titulu.

V Praze dne 29. 7. 2021

.....

podpis

## **Poděkování**

Chtěla bych tímto velice poděkovat Mgr. Lukášovi Brůhovi, Ph.D. za ujasnění směru práce a její vedení v průběhu celého zpracování, a za podnětné rady a doporučení. Dále mnohokrát děkuji Mgr. Matejovi Vrtichovi z ARCDATA PRAHA za ochotu a trpělivost při odpovídání na mé časté otázky a vysvětlování technologických řešení. Rovněž děkuji Mgr. Milanu Míkovi a Mgr. Jiřímu Maškovi za poskytnutí důležitých informací o dendrochronologických datech. Chci poděkovat i svojí rodině a přátelům za podporu, povzbuzení a modlitby, a dalším lidem, kteří mi při práci s něčím pomohli či poradili. A především bych chtěla poděkovat Bohu, bez jehož vedení a laskavé péče bych nedokázala tuto práci vůbec vytvořit.

## Abstrakt

Tato bakalářská práce se věnuje návrhu webové GISové aplikace určené pro zpracování dat o letokruhových chronologiích. Popisuje podrobný vývoj skriptu v jazyce Python, který ze vstupních dendrochronologických dat počítá mediány Cropperových hodnot dále vstupující do regresní funkce. Aproximované hodnoty získané regresní funkcí jsou následně interpolovány na území severní části chráněné krajinné oblasti Kokořínsko, určené pro tuto případovou studii. Dále je popisován postup publikace nástroje jako geoprocessingové služby na ArcGIS Server, a následné využití této služby v rámci přednastaveného vzorového widgetu. Přizpůsobený widget s připojenou geoprocessingovou službou je umístěn do webové aplikace vytvořené v prostředí ArcGIS Web AppBuilder. Aplikace je navržena tak, aby umožňovala prezentaci výsledků analýz geoprocessingové služby.

**Klíčová slova:** dendrochronologie, Cropperovy hodnoty, letokruhové chronologie, prostorová regrese, prostorová interpolace, webová aplikace, webový GIS, ArcGIS Pro, ArcGIS Enterprise, ArcGIS Server, geoprocessingová služba, webový nástroj, skript v jazyce Python, modul arcpy, Web AppBuilder, vlastní widget

## Abstract

This bachelor thesis concerns the design of a web GIS application tailored for tree-ring chronology data processing. It describes the development of a script tool in Python computing the medians of Cropper values from input dendrochronological data. The medians then enter a regression function. The approximated values obtained by the regression are afterwards interpolated within a region in the north of the Kokořínsko protected landscape area, designated for this case study. The procedure of publishing the tool as a geoprocessing service at an ArcGIS Server is described next, as well as its subsequent use within a preset sample widget. The customized widget with the geoprocessing service connected to it is then embedded into a web application created in the ArcGIS Web AppBuilder environment. The functionality of the application is designed to allow the presentation of geoprocessing service analysis results.

**Keywords:** dendrochronology, Cropper values, tree-ring chronologies, spatial regression, spatial interpolation, web application, web GIS, ArcGIS Pro, ArcGIS Enterprise, ArcGIS Server, geoprocessing service, web tool, Python script, arcpy module, Web AppBuilder, custom widget

# Obsah

<b>Seznam obrázků, tabulek a výpisů.....</b>	<b>9</b>
<b>Seznam zkratk.....</b>	<b>10</b>
<b>1 Úvod a cíle práce.....</b>	<b>11</b>
<b>2 Dendrochronologie.....</b>	<b>13</b>
2.1 Princip dendrochronologie.....	13
2.2 Souvislost s projektem TA ČR č. SS03010134.....	14
2.3 Analýza letokruhových chronologií.....	15
2.4 Cropperovy hodnoty.....	15
2.5 Regresní analýza.....	16
<b>3 Webová kartografie.....</b>	<b>17</b>
3.1 Webová mapová aplikace.....	17
3.2 Server.....	17
3.3 Mapový server.....	18
3.4 Model klient-server.....	18
3.5 Cloud computing.....	19
3.6 Hlavní jazyky webových stránek.....	19
3.7 Servisně orientovaná architektura.....	20
3.8 Webové mapové služby.....	20
3.9 Web AppBuilder.....	21
<b>4 Zdrojová data.....</b>	<b>22</b>
4.1 Dendrochronologická data.....	22
4.1.1 Metoda sběru dat.....	22
4.1.2 Vlastnosti dat letokruhových chronologií.....	22
4.1.3 Další obsah databáze.....	23
4.1.4 Umístění dat.....	23
4.2 Ostatní data potřebná pro analýzu.....	23
4.2.1 Rasterová data.....	24
4.2.2 Vektorová data.....	25
<b>5 Metodika.....</b>	<b>27</b>
5.1 Skript v jazyce Python.....	27
5.1.1 Parametry nástroje.....	28
5.1.2 Připojení databáze.....	29
5.1.3 Výběr dat.....	30
5.1.4 Výpočet Cropperových hodnot.....	32

5.1.5	Tvorba nové třídy prvků.....	34
5.1.6	Regresní funkce.....	35
5.1.7	Interpolační funkce.....	37
5.1.8	Reklasifikace.....	39
5.1.9	Symbologie.....	43
5.1.10	Ošetření výjimek.....	45
5.1.11	Log.....	46
5.2	Publikace geoprocessingové služby.....	47
5.2.1	Nastavení parametrů nástroje.....	47
5.2.2	Nastavení publikace.....	49
5.2.3	Oprávnění.....	49
5.3	Uživatelské rozhraní.....	50
<b>6</b>	<b>Výsledky.....</b>	<b>51</b>
<b>7</b>	<b>Diskuze.....</b>	<b>53</b>
<b>8</b>	<b>Závěr.....</b>	<b>56</b>
<b>9</b>	<b>Použité zdroje.....</b>	<b>57</b>
<b>10</b>	<b>Seznam příloh.....</b>	<b>66</b>



## Seznam obrázků, tabulek a výpisů

Obrázek 1: Stavba kmene s letokruhy.....	13
Obrázek 2: Schéma komunikace mezi mapovým serverem a uživatelem (klientem).....	18
Obrázek 3: Přehled vzorkovaných stanovišť a vymezení oblasti analýzy.....	24
Obrázek 4: Vzorec pro výpočet Cropperovy hodnoty.....	33
Obrázek 5: Princip metody nejmenších čtverců.....	36
Obrázek 6: Přehled definovaných parametrů.....	48
Obrázek 7: Příklad rasterového výstupu nástroje se symbologií nastavenou v ArcGIS Pro (predikce Cropperových hodnot) a bodů lokalit.....	51
Obrázek 8: Příklad reklasifikovaného rasteru interpolovaných Cropperových hodnot, imitující symbologii (v ArcGIS Pro) navrhovanou v Příloze 2, a bodů lokalit.....	52
Tabulka 1: Reklasifikace orientace.....	25
Tabulka 2: Vysvětlující proměnné a jejich zdroje.....	26
Tabulka 3: Reklasifikace rasteru s interpolovanými hodnotami mediánů Cropperových hodnot.	41
Tabulka 4: Stručný přehled reklasifikace rasteru.....	41
Výpis 1: Alternativa tvorby SDE connection file.....	29
Výpis 2: Alternativa tvorby remapRange.....	43
Výpis 3: Možné definování symbologie bodové třídy prvků.....	45

## Seznam zkratek

<b>2D, 3D</b> – dvou/třídídimenzionální	<b>PaaS</b> – Platform as Service
<b>API</b> – Application Programming Interface	<b>PřF UK</b> – Přírodovědecká fakulta University Karlovy
<b>ArcGIS</b> – Aeronautical Reconnaissance Coverage Geographic Information System	<b>REST</b> – Representational State Transfer
<b>C-hodnoty</b> – Cropperovy hodnoty	<b>RGBA</b> – Red-Green-Blue-Alpha
<b>CSS</b> - Cascading Style Sheets	<b>SaaS</b> – Software as Service
<b>CSV</b> – comma-separated values	<b>SDE</b> – Spatial Database Engine
<b>ČGS</b> – Česká geologická služba	<b>SFS</b> – Simple Fill Symbol
<b>ČHMÚ</b> – Český hydrometeorologický ústav	<b>SHP</b> – Shapefile
<b>ČR</b> – Česká republika	<b>SLS</b> – Simple Line Symbol
<b>DMR</b> – digitální model reliéfu	<b>SOA</b> – Service Oriented Architecture
<b>DMR 4G</b> – Digitální model reliéfu České republiky 4. generace	<b>SOAP</b> – Simple Object Access Protocol
<b>EPSG</b> – European Petroleum Survey Group	<b>SQL</b> – Structured Query Language
<b>ESRI</b> – Environmental Systems Research Institute	<b>TA ČR</b> – Technologická agentura České republiky
<b>GCS</b> – Geographic Coordinate System	<b>TSX</b> – TypeScript
<b>GIS</b> – geoinformační systémy	<b>TXT</b> – text
<b>GP</b> – geoprocessing	<b>UI</b> – user interface
<b>HTML</b> – HyperText Markup Language	<b>URL</b> – Uniform Resource Locator
<b>HTTP</b> – Hypertext Transfer Protocol	<b>WAB</b> – Web AppBuilder
<b>HTTPS</b> – Hypertext Transfer Protocol Secure	<b>WGS</b> – World Geodetic System
<b>CHKO</b> – chráněná krajinná oblast	<b>WKID</b> – 'Well-known ID'
<b>IaaS</b> – Infrastructure as a Service	<b>WMS</b> – Web Map Service
<b>JS</b> – JavaScript	<b>WYSIWYG</b> – What you see is what you get
<b>JSON</b> – JavaScript Object Notation	
<b>JSX</b> – JavaScript Extension	
<b>KB</b> – kilobyte	
<b>KFGG</b> – Katedra fyzické geografie a geoinformatiky	
<b>MB</b> – megabyte	
<b>OLS</b> – ordinary least squares	

# 1 Úvod a cíle práce

Mapy či mapě podobné produkty slouží člověku jako pomůcky při orientaci v prostoru už od nepaměti, a vznikaly s využitím rozličných materiálů a prostředků – od kostí a klů, kamene, přes hliněné destičky, papyrus a papír, až do dnešní doby digitalizace. S pokrokem vědy se objevily nové příležitosti umožňující využití digitálních technologií pro zpracování mapového díla, a ačkoliv má papírová podoba mapa své výhody a zajisté i určité kouzlo, digitální alternativa s sebou přináší možnosti, o kterých se dřív mohlo lidem jenom zdát. Z pohledu tvůrce může jít například o automatizaci, a tedy i zjednodušení početných úkonů potřebných pro vytvoření takového díla, pro uživatele naopak třeba o usnadnění přístupu k množství informací v něm obsažených. Rovněž přináší řadu benefitů všeobecná propojenost z informačního hlediska, rychlost sdílení dat po celém světě a dostupnost frekventovaných aktualizací datových zdrojů, a to vše zejména díky využití možností internetu. Obzvláště v poslední době, kdy dochází ke stále většímu rozšiřování webových geoinformačních systémů, se dostávají možnosti vlastní analýzy běžnému člověku stále více na dosah. V podstatě je možné považovat za typický příklad takové drobné uživatelské analýzy plánování výletu s využitím webových prohlížečů, kde je možnost nastavit určité parametry a omezení, a výsledkem bývá síťová analýza cesty, graf výškopisu a podobně. Právě tak mohou být a jsou využívány specializované nástroje a webové aplikace zaměřené na užší spektrum analytických a prezentačních potřeb v rámci různých odvětví.

Tato práce si klade za cíl navrhnout a implementovat webovou GISovou aplikaci, která bude sloužit potřebám Katedry fyzické geografie a geoekologie Přírodovědecké fakulty Karlovy univerzity v souvislosti s projektem Technologické agentury České republiky (TA ČR) (podrobněji v kapitole 2.2). Aplikace má umožňovat uživateli nadefinovat parametry výběru podmnožiny ze zdrojového datasetu letokruhových chronologií, který je poskytován zmíněnou katedrou, a také některých parametrů na ní prováděné prostorové analýzy. V závislosti na tom bude také prezentovat její výsledky předpovídající veličinu charakterizující hlavní lesní dřeviny v závislosti na údajích o potenciálně korelovaných jevech. Záměrem je vystavění této výsledné aplikace na databázi, jejíž návrh byl předmětem bakalářské práce H.-M. Majera v roce 2021 (Majer, 2021). Cílem tohoto projektu je tedy vytvořit za prvé geoprocessingovou (GP) část, která bude zpracovávat vstupní dataset, provede potřebné přípravné operace a výpočty a následně i samotnou regresní a interpolační funkci; za druhé pak propojení geoprocessingu s widgetem, který bude tvořit uživatelské rozhraní vytvořeného nástroje ve webové aplikaci, v rámci které

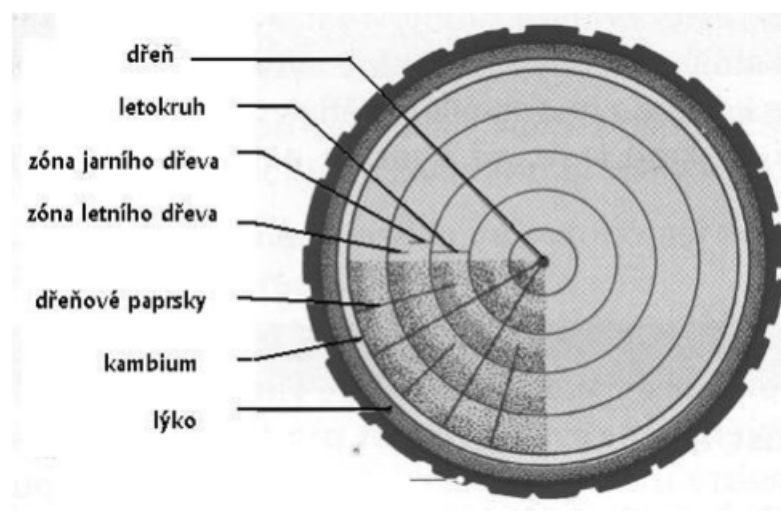
bude nadefinován. Výsledkem by mělo být zobrazení výstupní vrstvy analýzy online v mapovém okně aplikace a možnost stažení zvoleného výběru vstupních dat na disk uživatele. V závislosti na velikosti vstupního datasetu se bude tato případová studie omezovat pouze na část oblasti Kokořínsko, odkud výsledky vzorkování pocházejí. Po technologické stránce je při zpracování počítáno s využitím funkcí dostupných v GIS softwaru vyvíjeném společností ESRI, a to zejména ArcGIS Pro a Web AppBuilder (WAB). V rámci WAB budou využity poskytované widgety, jejichž funkčnost bude případně podle potřeby rozšířena.

## 2 Dendrochronologie

Aplikační oblastí této práce je dendrochronologie, v textu této kapitoly bude proto postupně představen princip dendrochronologie a možnost, jakým způsobem lze výstupy dendrochronologických měření analyzovat.

### 2.1 Princip dendrochronologie

Dendrochronologie je vědní obor věnující se sběru údajů získaných ze dřeva stromů a keřů, jejich kategorizaci vzhledem k prostorovému výskytu a časovému aspektu, a dále pak jejich zpracování a využití v navazujícím výzkumu a pro praktické účely. Vychází z předpokladu, že růst dřevin je ovlivňován vnějšími okolnostmi, jakými jsou např. klimatické podmínky, lidská činnost apod. (Sequens, 2007). Hlavním zdrojem používaných dat jsou letokruhy, které vznikají ve dřevinách rostoucích v oblastech se střídáním chladnějších a teplejších období (s nestálým podnebím); jeden letokruh odpovídá jednomu vegetačnímu období, což v našich zeměpisných šířkách koresponduje s obdobím jednoho roku, pokud nenastane výjimečná situace (např. disturbance), která by způsobila jeho nevytvoření (tzv. chybějící letokruh) nebo naopak vytvoření většího množství letokruhů (tzv. falešné letokruhy) (Wendland, 1975). Letokruhy jsou tvořeny kambiem, což je vrstva živých buněk dělivého pletiva, vykreslující tmavé prstence na průřezu kmene. Mezi vrstvami kambia se nachází světlejší a měkčí vrstva, která se vytváří v první části období s příznivými podmínkami, a tvoří široké buňky s tenkou stěnou; této vrstvě se říká jarní dřevo. V pozdější části příznivého (vegetačního) období vzniká tmavší, tvrdší vrstva silnostěnných zploštělých buněk – tzv. letní dřevo (Sequens, 2007). Vrstvy kmene s letokruhy jsou schematicky znázorněny na Obrázku 1.



Obrázek 1: Stavba kmene s letokruhy  
(Zdroj: ASPARA, 2015)

Primárním přístupem ke zpracování dendrochronologických dat je tzv. letokruhá analýza, která spočívá v aplikaci analytických metod, jejichž výsledkem může být datace zdrojového dřeva, detekce trendu růstu, identifikace vlivu určitého jevu, model predikcí a podobně (Sequens, 2007). Jedním z možných využití dendrochronologických metod je např. V rámci podoboru dendroarcheologie, jehož cílem je právě datace dřeva s patrnými letokruhy. Ze změřených a synchronizovaných letokruhových chronologií jsou vygenerovány průměrné křivky zvýrazňující společné extrémy. Ty jsou poté jsou porovnávány postupně od současnosti do minulosti se standardními chronologiemi, které existují specificky vždy pro konkrétní oblast a druh dřeviny, protože jedinci s těmito společnými vlastnostmi vykazují zpravidla velmi podobné skladby přírůstků, což v ideálním případě vede např. k dataci zdrojového dřeva (Dendroarcheologie, 2021).

Další součástí oblasti dendrochronologie je dendroekologie, která se zaměřuje na letokruhové chronologie v souvislosti s ekologickými problémy a otázkami týkajícími se životního prostředí. Mezi její podobory patří například dendrohydrologie, dendroglaciologie a také dendroklimatologie, jejímž cílem je rekonstrukce a výzkum změn klimatu a jeho vlivu na letokruhové přírůstky apod. (Dendroekologie, 2021; Sequens, 2007).

## **2.2 Souvislost s projektem TA ČR č. SS03010134**

Právě problematikou klimatických změn se v dnešní době zabývá řada projektů, které jsou zaměřeny na studium klimatických, hydrologických a dalších vlivů na životní prostředí pojmících se k těmto změnám. Mezi jejich cíle patří formulace stanovisek, která by vedla k omezení případného negativního vlivu a rizikových stavů v budoucnosti. Jedním z takových projektů je projekt s názvem „Databáze letokruhových chronologií jako nástroj pro evidenci a predikci reakce hlavních lesních dřevin na klimatickou změnu“ (SS03010134) zahájený v roce 2021 a řešený na Karlově univerzitě s podporou Technologické agentury České republiky (TA ČR).

Cílem tohoto projektu je sestavení celorepublikové geodatabáze letokruhových chronologií a dalších vlastností měřených lokalit; tato databáze pak bude zdrojem pro analýzy, jejichž výstupem mají být mapové vrstvy a další výstupy charakterizující vliv klimatu na vybrané převažující dřeviny v současnosti i predikující tento vliv do budoucnosti. Výsledky mají být využity pro posuzování vlivu klimatické změny na ekosystémy a jejich fungování, a pro navrhování budoucích a hodnocení dosavadních mitigačních opatření (s cílem zmírnit dopady

změn klimatu). Zejména jde o evaluaci vztahu mezi klimatem a letokruhovými přírůstky, charakterizovanými již zmíněnými letokruhovými sériemi, které jsou v zadání projektu definovány jako „přímo měřené charakteristiky ročního přírůstu kmenové biomasy dřevin“ (TA ČR, 2021, s.14), a predikci budoucích změn v produkci na území České republiky na základě tohoto vztahu (TA ČR, 2021).

### 2.3 Analýza letokruhových chronologií

Důležitým aspektem analýzy letokruhových chronologií jsou tzv. „event years“, které Schweingruber et al. (1990) definuje jako roky vyznačující se tím, že jsou oproti ostatním letům v omezeném okolním rozmezí letokruhové řady znatelně vyčnívající, a také tzv. „pointer years“, vymezené jako roky, ve kterých se vyskytuje „event year“ u většího počtu stromů s analyzovanou chronologií. (Volba hranice určující, zda je hodnota v určitém roce význačná oproti ostatním, je na uvážení zpracovatele, ale existují doporučené rozsahy, kde by se tato hranice měla pohybovat.) Studium takovýchto růstových anomálií se zabývá řada odborných prací. V rámci průzkumu popsaného Jetschke et al. (2019), zahrnujícího 121 publikací z let 1998 až 2018 zabývajících se hledáním pointer years, byly zjištěny čtyři pro tento účel nejčastěji používané metody ve zmíněném období:

- interval trend (intervalový trend),
- relative growth change (relativní změna růstu),
- normalization in a moving window (normalizace v pohyblivém okně) a
- extreme values of chronology (extrémní hodnoty chronologie).

Z těchto metod byla ve zkoumaných publikacích nejčastěji využita metoda normalizace v pohyblivém okně, jinak též Cropperova metoda. Jejím výstupem je série tzv. Cropperových hodnot (Jetschke, van der Maaten, van der Maaten-Theunissen, 2019). Normalizace probíhá v rámci každé letokruhové chronologie zvlášť, a výsledkem je vždy jedna chronologie Cropperových hodnot (dále jako C-hodnoty) z každé chronologie přírůstků nebo jejich přepočítaných hodnot; princip této metody je podrobně popsán v podkapitole 5.1.4.

### 2.4 Cropperovy hodnoty

C-hodnoty „odrážejí proměnlivost z hlediska směrodatných odchylek od lokálního průměru“<sup>1</sup> (Jetschke, van der Maaten, van der Maaten-Theunissen, 2019, s.56, překlad autorky). Jinými slovy, C-hodnota ukazuje, jak moc je daný rok extrémní v rámci časového okna (plovoucího

---

1 „Cropper values reflect the variations in terms of standard deviations from the local mean“

okna) se zvoleným rozsahem. Její kladnost pak odráží relativní nárůst letokruhového přírůstku, zatímco zápornost indikuje relativní pokles. Velikost absolutních hodnot C-hodnot značí míru intenzity této změny, což vede k důležitosti pointer years<sup>2</sup>.

## 2.5 Regresní analýza

Podle různých velikostí C-hodnot na vzorkovaných stanovištích lze poté analyzovat předpokládané souvislosti s klimatickými, geomorfologickými, hydrologickými, antropogenními a dalšími možnými vlivy. Tyto souvislosti mohou být modelovány s využitím regresního modelu, jehož cílem je vysvětlení proměnlivosti chování vysvětlované (závislé) proměnné na vysvětlující (nezávislé) proměnné/proměnných. Výsledkem může být pak důkaz (potvrzení) závislosti sledované veličiny na těchto proměnných, a posléze je na základě této korelace možné také aproximovat střední hodnotu této veličiny v lokalitách, kde neproběhlo měření, ale jsou v nich dostupné hodnoty vysvětlujících proměnných (faktorů které mají vliv na sledovanou veličinu) (Zvára, 2013).

V této práci je využita aproximace v síti neměřených lokalit pomocí regresní funkce a následná interpolace mezi odhadovanými (středními) hodnotami. Ačkoliv se dendroklimatologie i zmíněný projekt TA ČR zaměřují na závislost letokruhových přírůstků na vlivu klimatu, data poskytnutá pro tuto bakalářskou práci v současnosti nezahrnují údaje o klimatických charakteristikách; proto byla analýza zaměřena na obecně geografické charakteristiky. Při nahrazení původních dat klimatickými daty nebo při rozšíření o ně ale může být metoda použita analogicky k vysvětlení a predikci vysvětlované proměnné podle kteréhokoliv datasetu, jehož rozsah bude dostatečný pro požadované území predikce.

---

2 Mgr. Milan Mík, e-mailové sdělení 11. 5. 2021



### 3 Webová kartografie

Webová kartografie je odvětví kartografie specifické tím, že se omezuje na prostředí webu. Její součástí je příprava vedoucí k vytvoření webové mapy, samotná tvorba mapy, i následné používání mapového výstupu (Nétek, 2020). Webová mapa je pak podle Sacka (2017) taková mapa, která je, typicky v rámci webové stránky, zveřejněna a zpřístupněna prostřednictvím internetu. Dokumentace ArcGIS Online ji zase definuje v kontextu prostředí ArcGIS jako „interaktivní zobrazení geografických informací“<sup>3</sup> (ESRI 2021an). Nicméně webová mapa může být dvojího druhu – buď statická, kdy zůstávají obrazová data stejná a vizuální obsah jako takový není ovlivněn případným uživatelským zásahem, nebo dynamická, která se může měnit v závislosti na manipulaci uživatelem, nebo je její obsah tvořen animací (přednastavené postupné pravidelné změny) (Sack, 2017). Proces webového mapování (web mapping) potom Neumann (2017) definuje jako „navrhování, implementac[i], vytváření a předávání map na World Wide Webu“<sup>4</sup>(překlad autorky), a dále podotýká, že se týká zejména problematiky technologických řešení, oproti webové kartografii, věnující se komplexněji i teoretickému hledisku webové mapové tvorby.

#### 3.1 Webová mapová aplikace

Webová mapová aplikace je v kartografické souvislosti na webu umístěná aplikace, jejímž smyslem je prezentace mapových výstupů a dalších kartografických materiálů. Uživatel má také často možnost do různé míry se zobrazeným obsahem interagovat. Příkladem mohou být třeba běžně používané Mapy.cz, nebo také výstupy orgánů a společností, jako např. geovědní mapy ČGS nebo mapa rozvodnic ČHMÚ, využívající aplikační platformu Esri. Aplikace jako taková zpravidla zahrnuje tři základní prvky, kterými je webový server, mapový server a rozhraní, pomocí kterého jsou spravována data (Nétek, 2020).

#### 3.2 Server

Serverem se z fyzického hlediska označuje počítač, zpravidla o vysoké výkonnosti, který umožňuje dalším zařízením v síti, aby se k němu připojily a využily jeho služeb a kapacity (IT Slovník, b.r.). Z druhého pohledu jde pak o software na jednom či více počítačích, který komunikuje se vzdálenými zařízeními a předává jim informace uložené na zařízení, na kterém je nainstalován (Sack, 2017). Existují různé typy serverů, včetně např. souborového, tiskového

---

3 „interactive display of geographic information“

4 „designing, implementing, generating and delivering maps on the World Wide Web“

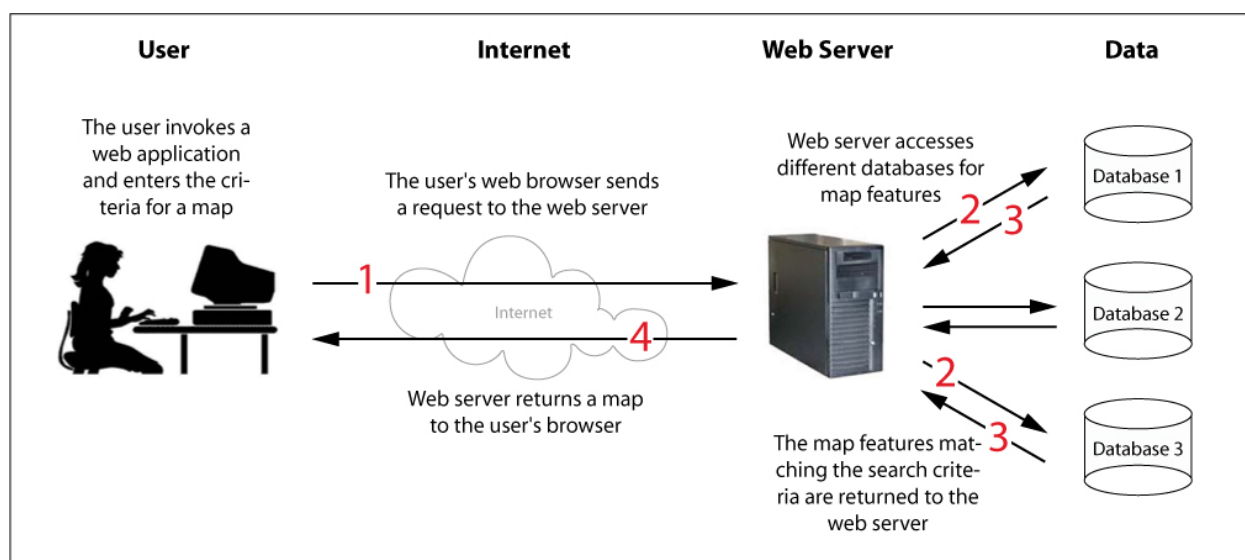
nebo nejběžněji používaného webového; jedním z typů serveru je i aplikační server (Nétek, 2020), který je specificky určen pro spravování provozu sdílených aplikací (ManagementMania, 2020). Jedním z příkladů aplikačního serveru je také je mapový server.

### 3.3 Mapový server

Mapový server je zvláštní typ serveru, který slouží ke „publikování geoprostorových webových služeb“<sup>5</sup> (Sack, 2017). Mapovým serverem je například ArcGIS Server, GeoServer, OpenMapTiles apod. V souvislosti s mapovým serverem je důležitým prvkem mapový portál. Podle Néteka (2020, s.27) hraje mapový portál roli „rozcestníku“ ze kterého lze přistupovat k jednotlivým mapovým výstupům, aplikacím apod. Slouží tedy jako uživatelské rozhraní serveru (webová stránka), které zpřístupňuje informace a služby ze serveru uživatelům. Typickým příkladem může být v souvislosti s touto prací Portal for ArcGIS, jehož součástí je např. aplikace pro sestavování aplikací Web AppBuilder (WAB).

### 3.4 Model klient-server

Základem fungování nejen webových stránek je komunikace mezi serverem a tzv. klientem – architektura klient-server. V rámci tohoto systému je tedy umožněna síťová komunikace, založená na posílání a přijímání informací mezi serverovým programem počítače, který tento obsah uchovává, a počítačem s klientským softwarem, který vysílá směrem k serveru požadavky (Sack, 2017). Schéma modelu klient-server je zachyceno na Obrázku 2. Komunikace probíhá



Obrázek 2: Schéma komunikace mezi mapovým serverem a uživatelem (klientem)  
(Zdroj: Duggan, 2014)

5 „A specialized web server used to publish geospatial web services.“

s využitím protokolů. Pro webové stránky je používán protokol HTTPS (Hypertext Transfer Protocol Secure), případně starší, nezabezpečený protokol HTTP; tyto protokoly jsou určené pro přenos dokumentů ve formátu HTML (existují i další protokoly pro jiné účely, např. POP3 pro elektronickou poštu, nebo FTP pro přenos souborů) (Nétek, 2020).

### 3.5 Cloud computing

Cloud computing je způsob, jakým lze skrze internetové zprostředkování využít kapacity výpočetní techniky, která se nachází mimo lokální zařízení uživatele (Nétek, 2020). Margaritis (2011, s.viii) definuje jeho účel mimo jiné jako „realizaci síťového přístupu ke sdíleným výpočetním zdrojům (např. sítím, serverům, datovým úložištím, aplikacím a službám)“. Rovněž zmiňuje podmínku rychlého a stálého přístupu ke sdíleným prostředkům. Typy cloud computingu lze rozlišovat například podle míry konfigurovatelnosti využití cloudu a velikosti výpočetního výkonu uživatelem od modelu IaaS (Infrastructure as a Service) s největší možností přizpůsobení přes možnost spravovat platformu dle modelu PaaS (Platform as Service) až k modelu SaaS, kdy je placeným produktem je čas využití poskytovaného software nebo aplikace (Nétek, 2020). Mezi příklady služeb, které tento poslední model využívají, patří například Dropbox, Canva, Lumen5 (Fetcher, 2021) nebo také ArcGIS Online.

### 3.6 Hlavní jazyky webových stránek

V současnosti jsou pro tvorbu webových stránek typicky používány tři základní jazyky: HTML (HyperText Markup Language), CSS (Cascading Style Sheets) a JS (JavaScript). HTML, CSS i JS jsou front-endové client-side technologie, což znamená, že jejich kód je vykonáván na straně klienta a společně vytváří tu část webových stránek, která se zobrazuje uživateli (Uchiha, 2018; IT Slovník, b.r.).

**HTML** je značkovací jazyk, který tvoří hlavní strukturu webové stránky, určuje jednotlivé prvky na stránce. Jeho současná specifikace HTML5 poskytuje např. možnost tvorby offline aplikací, pouštění multimediálních prvků v rámci prohlížeče, nastavení stylů se odděluje mimo základní kód stránky do souboru v CSS3 (viz níže) a je možné specifikovat i vlastní atributy elementů. Důležité je že tato verze podporuje některé prvky pokročilé grafiky, využitelné například právě pro vykreslování 2D i 3D map (VA Vyškov, b.r.; Nétek, 2020).

CSS (kaskádové styly) je jazyk umožňující formátování stylů stránek nejen v HTML. Jeho specifikace CSS3, která je aktuální, zprostředkovává navíc mimo jiné i vizualizační aspekt znázornění mapových prvků podle kartografických metod (Štráfelda, b.r.; Néték, 2020).

JS je objektově orientovaný programovací jazyk, který zprostředkovává interaktivitu HTML a interpretuje ho přímo prohlížeč (Néték, 2020). Je možné ho použít i na straně serveru (Štráfelda, b.r.).

### **3.7 Servisně orientovaná architektura**

Servisně orientovaná architektura (SOA) je obecný způsob, jakým lze přistupovat k sestavování (např. aplikačních) struktur pomocí jednotlivých služeb, nástrojů apod., jejichž integraci zajišťuje vysokoúrovňové rozhraní, skrze které je každá zapojená služba k dispozici (Leština, 2007); zároveň je tak umožněn dálkový přístup k datům a nástrojům (Néték, 2020). Jednotlivé komponenty na sebe nejsou navázány pevnými vztahy, naproti tomu jsou služby definovány svými metadaty, která určují, jaké má daná služba parametry, vstupy a výstupy a podobně (Néték, 2020). Takovou službu nebo jiný komponent je tedy možné jednodušeji nahradit či opravit, a také ji díky takto volnému spojení v rámci určité aplikační struktury lze použít ve struktuře jiné (Hao, 2003; IBM Cloud Education, 2019), v závislosti na již zmíněných parametrech.

### **3.8 Webové mapové služby**

Webové mapové služby (WMS, web map service(s)) vycházejí z konceptu SOA a fungují na základě modelu komunikace klient-server. Použití služby je z klientské strany vyvoláno zadáním URL adresy dané služby a potřebných vstupů, zpátky se pak vrací výsledek proběhnutí služby. Existují různé schválené služby a protokoly pro přístup k jednotlivým processingovým službám, včetně například služby SOAP, nebo REST, což je využíváno společností Esri pro systém ArcGIS Serveru. Aplikační programovací rozhraní (API) REST je komunikační rozhraní se serverovým programem (velmi vhodné pro WMS, ale používané i v dalších webových službách), které z větší části, i když ne zcela, nahradilo předcházející SOAP (Mason, 2011). Rozhraní přístupu k této službě je uživatelsky přívětivé. Restové služby většinou využívají JSON formát pro odpovědi (komunikaci) (Holec, 2021). O rozhraní REST píše Pichlík ve svém článku následující: „Representational State Transfer (REST) je koncept pro design distribuované architektury. Distribuovaná architektura v tomto smyslu znamená, že části programu běží na

různých strojích a pro svojí komunikaci využívají síť.“ (Pichlík, 2007). Jako příklad takového programu autor dále uvádí např. právě webovou aplikaci, v rámci které probíhá komunikace internetového prohlížeče s webovým serverem.

### 3.9 Web AppBuilder

Jednou z cest, jakými dosáhnout vytvoření aplikace, která může zahrnovat geoprocessingovou službu způsobem, jakým bylo popsáno v předchozích kapitolách, je využití Web AppBuilderu (WAB). Web AppBuilder je aplikace typu WYSIWYG (what you see is what you get, tedy rovnou zobrazující podobu toho, co je konfigurováno), která je specializována na účel tvorby webových aplikací. Aplikaci lze sestavit prostřednictvím widgetů, což jsou nástroje zprostředkávající jednotlivé funkcionality aplikace. S využitím vývojářské edice (Developer Edition) je možné rovněž vytvořit vlastní nový widget nebo přizpůsobit některý stávající či vzorový widget ESRI 2021<sup>6</sup>. Nejpřímější způsob, jakým zpřístupnit vytvořenou GP službu, je za využití konkrétního předpřipraveného widgetu jménem Geoprocessing.

Nevýhodou takovéto vysokoúrovňové sestavující aplikace je méně prostoru pro libovolné přizpůsobení výsledné aplikace. Výhodou je naopak rychlost a jednoduchost celého pracovního postupu v porovnání s řadou jiných možností vývoje.

WAB je starší technologií, jejíž implementace v JavaScriptu je závislá na JS knihovně Dojo Toolkit. Předlohy widgetů využívají tzv. Dojo Template-based Widgets, oproti v modernějších JavaScriptových aplikacích používanému JSX/TSX<sup>6</sup> (tj. JavaScript Extension/TypeScript). Mezi další a také novější možnosti tvorby webových aplikací pomocí alternativ produkovaných společností Esri patří například použití aplikace ArcGIS Experience Builder jakožto nástupce Web AppBuilderu, nebo naprogramování aplikace od základů s využitím ArcGIS API for JavaScript.

---

<sup>6</sup> Mgr. Matej Vrtich, e-mailové sdělení 13. 7. 2021

## 4 Zdrojová data

Data využitá v rámci zpracování této práce se dělí na dendrochronologická data poskytnutá KFGG pro účely analýzy, a dále pak data nezbytná pro uskutečnění analýzy, která ale nejsou středem zájmu tohoto projektu.

### 4.1 Dendrochronologická data

Data pro tuto případovou studii, nad kterými byla navrhována aplikace, pochází z měření letokruhových chronologií druhu borovice lesní (*Pinus sylvestris*), v databázi označeného jako „PISY“. Měření proběhlo v roce 2020, kromě dvou stanovišť, která byla vzorkována v letech 2017 a 2018 – lokalita KOS (pův. KOS\_S) a KOJ (pův. KOS) (viz Mík, 2020). Měřeno bylo ve 20 různých lokalitách v severní části Kokořínska mezi Českou Lípou a Štětím, v okolí vrchu Kostelec; měřené lokality se nacházejí na různých částech reliéfu.

#### 4.1.1 Metoda sběru dat

Poskytnutá testovací data byla získána ručním měřením, které probíhalo podle následující metodiky. Speciálním dutým nástrojem, tzv. přírůstovým (Presslerovým) nebozezem je přibližně ve výšce hrudi vyvrtán úzký válec dřeva, procházející kolmo přes letokruhy. Tento vzorek je vysušen a připraven pro měření přírůstků, které probíhá poloautomaticky. Počítačový program detekuje jednotlivé letokruhy a výsledek je poté zkontrolován zpracovatelem, který se přesvědčí, že detekce všude proběhla správně, a případné chyby ručně opraví (Mík, 2020). (Toto platí pro dostupná testovací data, která byla k dispozici pro návrh aplikace.)

#### 4.1.2 Vlastnosti dat letokruhových chronologií

Zdrojová databáze obsahuje sadu surových (raw) dat, a rovněž tatáž data detrendovaná metodou spline křivky s délkou okna 50 let, kromě dvou stanovišť, která nemají detrendovanou verzi. K jednotlivým záznamům velikostí přírůstků patří údaj o roku, ke kterému náleží. Všechny chronologie jsou kontinuální a žádná z nich nekončí dříve než začíná jiná. Většina záznamů ze stanovišť zahrnuje 26 vzorků, čtyři z lokalit ale větší množství, vzhledem k tomu, že zde byly vzorkovány všechny stromy s průměrem větším než 10 cm. Z některých stromů bylo odebráno více vzorků.

### 4.1.3 Další obsah databáze

V databázi jsou uloženy za jednotlivá stanoviště tyto údaje: přiřazený unikátní kód lokality, dále pak název, měřený druh stromu, hloubka kořenů, typ půdy, zeměpisné souřadnice v systému WGS 1984 (ve stupních) a nadmořská výška. Ačkoliv je pro ně vyhrazeno pole, není zde vyplněn sklon ani orientace, a proto bylo nutné tyto hodnoty do analýzy doplnit z jiného zdroje. V databázi jsou pro nadmořskou výšku použity hodnoty DMR 4G, v této bakalářské práci byly v geoprocessingu použity hodnoty z rasteru DMR z ArcČR (vycházející ze stejného zdroje), a to pro nadmořskou výšku i z ní vycházející sklon a orientaci. Na úrovni stromů se v databázi nachází údaje o průměru stromu ve výšce hrudi (DBH), výšce stromu, výšce nasazení koruny, poloměru koruny, kompetičním indexu a šířce bělového dřeva. Stromy z několika lokalit mají uvedenu i svou vlastní nadmořskou výšku, sklon a orientaci, případně i vlastní souřadnice (Majer, 2021).

### 4.1.4 Umístění dat

Data jsou umístěna v relační databázi, jejíž návrh, tvorba a naplnění daty byly předmětem bakalářské práce H.-M. Majera (Majer, 2021). Databáze byla nasazena na fakultní server DATAMAP a zpřístupněna přes přístupové údaje pro použití jako datový zdroj v rámci této bakalářské práce.

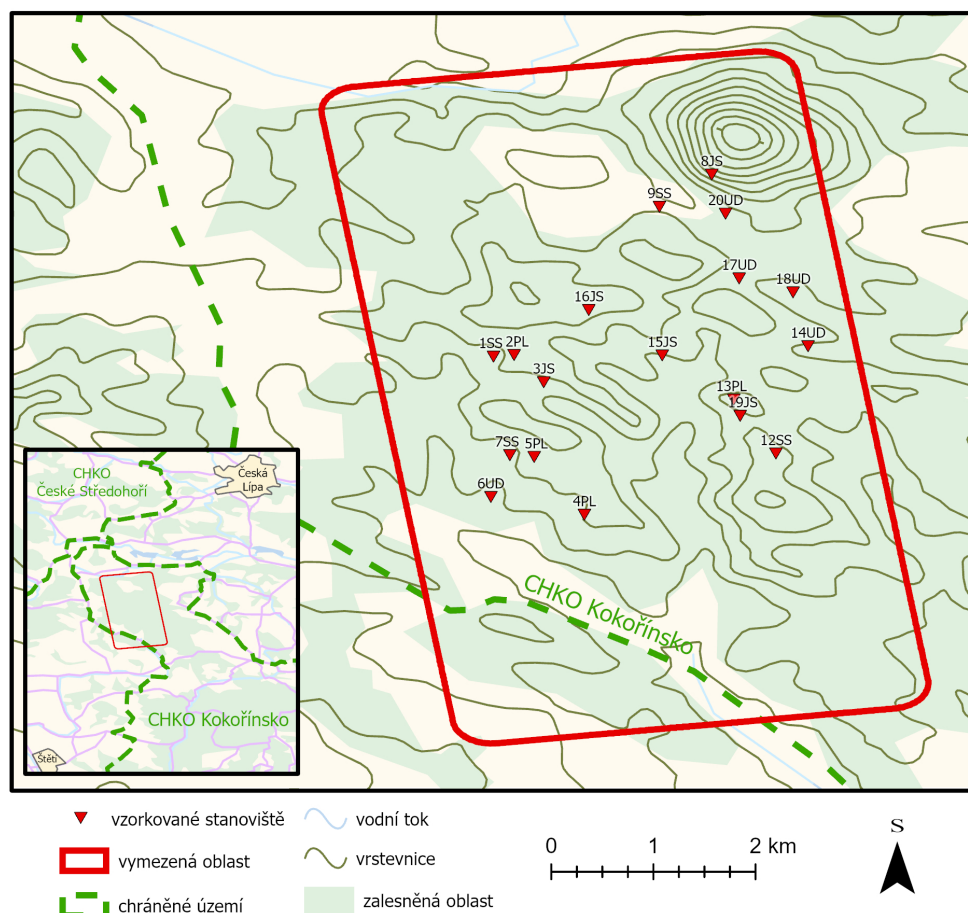
## 4.2 Ostatní data potřebná pro analýzu

Pro účely analýzy bylo nezbytné, aby byla připravena bodová vrstva, do které budou regresní funkcí aproximována data, a mezi těmito body pak mohla proběhnout interpolace. Aby mohl výpočet regresní funkce proběhnout, musí body obsahovat pole s vysvětlujícími proměnnými; tyto informace se v případě této studie získají převážně z rasterových podkladů. K nim i k bodové vrstvě je přístupováno přes webové rozhraní serveru Portal for ArcGIS. Pro komunikaci s ArcGIS Enterprise z ArcGIS Pro je potřeba nejprve ve správě portálů ArcGIS Pro přidat adresu Portalu, nastavit ho jako aktivní a přihlásit se serverovými přihlašovacími údaji (Holubec, 2017).

Prostorové vrstvy jsou zpracovány v geografickém souřadnicovém systému WGS 1984, který má kód EPSG (WKID) 4326. Vstupní data mající zdroj mimo dendrochronologickou databázi jsou popsána v následujících podkapitolách.

#### 4.2.1 Rasterová data

Rasterové podklady byly potřebné pro získání vysvětlujících charakteristik pro bodovou vrstvu predikovaných lokalit i pro bodovou třídu prvků reprezentujících měřená stanoviště, protože kromě nadmořské výšky tyto údaje nejsou v databázi kompletní. Raster nadmořských výšek pochází z digitální geodatabáze ArcČR 500 v.3.3; je odvozen na základě Digitálního modelu reliéfu České republiky 4. generace (DMR 4G), který „je určen k analýzám terénních poměrů regionálního charakteru a rozsahu, např. při (...) modelování přírodních jevů“ (ČÚZK, 2010). Raster charakterizující nadmořskou výšku byl transformován ze systému S-JTSK\_Krovak\_East\_North do GCS\_WGS\_1984 (funkce Project Raster); z něj pak vycházejí rastery sklonu terénu a orientace plochy terénu, které z něj byly odvozeny pomocí funkcí Slope a Aspect v programu ArcGIS Pro. Raster orientace byl pro potřeby studie navíc překlasifikován na čtyři hlavní směry (a případnou rovinu), jak je patrné z Tabulky 1. Všechny tři rastery jsou pro účely této případové studie oříznuty jen pro část oblasti Kokořínsko, omezenou polygonem obdélníkového tvaru (viz Obrázek 3).



Obrázek 3: Přehled vzorkovaných stanovišť a vymezení oblasti analýzy (Zdroj: autorka, datové vrstvy z ArcČR 500 v.3.3)



Rasterové podklady byly sdíleny na server, aby k nim mohl mít přístup kdokoliv z organizace. Jako typ sdílené vrstvy byla zvolena Imagery layer (vrstva obrazových dat), která podporuje funkcionality potřebné pro prostorovou analýzu; lze nad ní tvořit dotazy a provádět analýzy (ESRI, 2021a). Bylo zadáno zkopírování všech dat na server. Po doplnění popisu, tagů a metadat a zvolení cílové složky byla vrstva validována (analyzována) a publikována na server. Publikovat lze s oprávněním publikace server-based layers, na což postačuje výchozí role Publisher (ESRI, 2021b).

Od původní hodnoty	Do původní hodnoty	Nová hodnota	Světová strana
0	45	1	sever
45	135	2	východ
135	225	3	jih
225	315	4	západ
315	360	1	sever
-1	-1	-1	žádná (rovina, nulový sklon)

*Tabulka 1: Reklasifikace orientace  
(Zdroj: autorka)*

#### **4.2.2 Vektorová data**

V programu ArcGIS Pro byla vytvořena bodová vrstva s body umístěnými na průsečících pravidelné čtvercové sítě o velikosti buňky 100 × 100 metrů. Následně byla oříznuta na část území Kokořínska, určené pro tuto případovou studii, a pomocí nástroje Project byla promítnuta do souřadnicového systému GCS\_WGS\_1984. Do atributové tabulky byly k bodům doplněny souřadnice zeměpisné šířky a délky a pole s nimi přejmenována na „latitude“ a „longitude“, aby byly názvy mezi materiály sjednoceny. Následně byla přidána i pole se zbývajícími vysvětlujícími proměnnými – nadmořskou výškou, sklonem a orientací terénu; tyto hodnoty byly získány ze stejných rasterových podkladů, které byly následně sdíleny na server (jak bylo zmíněno v podkapitole 4.2.1). Výsledná bodová vrstva tedy obsahuje všechny případně potřebné hodnoty vysvětlujících proměnných (viz Tabulka 4)

Údaj	Jednotky	Zdroj	Poznámka
nadmořská výška (elevation)	metry (m)	raster - digitální model reliéfu (DMR)	v Baltském výškovém systému po vyrovnání (Bpv)
sklon (slope)	stupně (°)	raster sklonu vytvořený z DMR	
orientace vzhledem ke světovým stranám (aspect)	-	raster orientace vytvořený z DMR (planární metodou)	z 360° hodnot ve směru hod. r. reklasifikováno dle tabulky níže
zeměpisná šířka (latitude)	stupně (°)	přidání souřadnice Y ve zvoleném souřadnicovém syst.	souřadnicový systém: GCS_WGS_1984
zeměpisná délka (longitude)	stupně (°)	přidání souřadnice X ve zvoleném souřadnicovém syst.	souřadnicový systém: GCS_WGS_1984

*Tabulka 2: Vysvětlující proměnné a jejich zdroje  
(Zdroj: autorka)*

Poznámka: Vzdálenosti pro zeměpisné souřadnice jsou analyzovány pomocí chordálních vzdáleností v metrech.

Aby mohla být bodová vrstva přístupná pro členy organizace, byla namísto pevného disku umístěna na server. Z tabulky obsahu proběhlo sdílení vytvořené vrstvy jako web layeru – byl doplněn popis a tagy a zadáno, že mají být všechna data zkopírována na server. Jako typ sdílené vrstvy byla zvolena (Hosted) Feature layer (tedy hostovaná vrstva prvků), což je vrstva vektorových dat, která „byla umístěna do portálu, a tedy jsou hostována portálem a jsou zpřístupněna prostřednictvím ArcGIS REST API webové služby, konkrétně Feature service.“ (Vrtich, 2018, s. 34) Tento typ vrstvy podporuje dotazování na data, jejich zobrazení i úpravy (ESRI, 2021c) a je podporováno i jeho využití jakožto prvku vstupujícího do analýz (ESRI, 2021d). Všechna data byla zkopírována na server. Dostupnost vrstvy byla nastavena na sdílení se všemi členy organizace (tedy v rámci GIS serveru geografické sekce PřF UK). Po validaci a vyřešení chyb a varování byla vrstva publikována na server (ARCDATA, 2017). Publikovat lze s oprávněním publikovat hosted web layers, což je rovněž umožněno ve výchozí roli Publisher. (ESRI, 2021d) ve skriptu může být na tuto vrstvu odkázáno pomocí URL přes službu REST API pod indexem 0 (ESRI, 2021e).

## 5 Metodika

Pro vytvoření webové aplikace bylo zapotřebí napsat skript provádějící potřebný geoprocessing, publikovat ho jako službu na server, a s touto službou poté propojit a nastavit widget, který k ní zajistí přístup uživateli. Pak bylo potřeba nadefinovat novou aplikaci v sestavovací aplikaci Web AppBuilder a tento widget do ní vložit. Následující kapitoly popisují průběh a odůvodnění jednotlivých kroků tohoto pracovního postupu.

### 5.1 Skript v jazyce Python

Pro sestavení skriptu bylo zásadní použití modulu `arcpy`, který umožňuje přístup k souborům, vrstvám a nastavením projektu ArcGIS Pro a manipulaci s nimi pomocí funkcí a nástrojů poskytovaných tímto software. Vývoj kódu probíhal v editačním prostředí VisualStudioCode. Pro možnost využití modulu `arcpy` byl nastaven interpret Pythonu, verze Pythonu 3.6.10, poskytnutá v rámci balíčku ArcGIS Pro (conda environment).

Použité Python moduly: `arcpy`, `os`, `tempfile`, `shutil`, `math`, `statistics`, `collections`, `sys`, `time`

V rámci modulu `arcpy`:

- Použité ArcGIS moduly:
  - `da` – Data Access (ESRI, 2021f)
- Použitá ArcGIS rozšíření:
  - `sa` – Spatial Analyst (ESRI, 2021g)
  - `ga` – Geostatistical Analyst (ESRI, 2021h)
- Použité ArcGIS toolbox:
  - `management` – Data Management (ESRI, 2021i)
  - `conversion` – Conversion (ESRI, 2021j)
  - `stats` – Spatial Statistics toolbox (ESRI, 2021k).

Testování skriptu probíhalo za využití prostředí `scratchWorkspace`, jehož definování bylo před publikací odstraněno, protože server si poté nastaví prostředí a umístění dat na něž skript odkazuje pomocí proměnných `scratchFolder` a `scratchGDB`<sup>7</sup>. Souborové mezivýstupy a výstupy jsou zapisovány do umístění `arcpy.env.scratchFolder`, které odkazuje na složku, ostatní jsou zapisovány do umístění `arcpy.env.scratchGDB`, které odkazuje na souborovou geodatabázi. Pro obě tato umístění platí, že jsou pro ukládání při geoprocessingu vždy k dispozici (s právem

---

<sup>7</sup> Mgr. Matej Vrtich, e-mailové sdělení 10. 5. 2021

zápisu do nich), a když nejsou nalezena existující, jsou podle potřeby při běhu skriptu vytvořena (ESRI, 2021l; ESRI, 2021m).

Ukládání do paměti „in\_memory“ nebylo ve skriptu využito, protože se pracuje s objemnými daty (paměť „in\_memory“ je vhodná pro ukládání dat v řádu MB/KB) a navrhovaná služba je asynchronní, a je tedy potřeba, aby byly výsledky geoprocessingu zapsány na disk, odkud pak bude pro uživatele možné jejich stažení<sup>8</sup>. Soubory z paměti navíc není možné uložit pro vykreslení mapovou službou a při zavření aplikace jsou vždy odstraněny. Kdyby neexistovaly tyto protiargumenty, výhodou použití paměti „in\_memory“ by byla zejména rychlost oproti zápisu na disk (ESRI, 2021n)

Sestavení relativních cest bylo uskutečněno pomocí modulu „os“ metodou `os.path.join()`, která umožňuje sestavení cesty k souboru po částech, přičemž to, že se potřebný řetězec v kódu nenachází v jednom řetězci, zabraňuje chybám v důsledku interpretace spojovacího znaku jako znaku se speciálním významem. Zároveň platí, že „vzhledem k tomu, že různé operační systémy mají odlišné konvence pro psaní cest, existuje několik verzí tohoto modulu v rámci základní knihovny. Modul `os.path` je vždy vhodný pro ten operační systém, na kterém Python běží, a tedy použitelný pro lokální umístění“<sup>9</sup> (Python, 2021, překlad autorky). Existují ale i další moduly, které cesty sestavují specifickými způsoby.

V následujících podkapitolách je popsán průběh a princip fungování napsaného skriptu.

### 5.1.1 Parametry nástroje

Vstupní parametry jsou volány pomocí příkazu `GetParameterAsText(x)`, kde „x“ značí index, který identifikuje konkrétní parametr mezi ostatními parametry skriptu. Funkce vrací parametr zadaný uživatelem ve formě řetězce. Výstupní parametry jsou definovány příkazem `SetParameterAsText(x,s)`, kde „x“ opět značí identifikační index parametru (indexování je společné pro vstupy i výstupy) a „s“ řetězec názvu výstupu (např. umístění výstupního souboru). Tato funkce je použita v první řadě pro určení výstupních vrstev analýzy, ve druhé řadě za účelem výpisu zpráv informujících uživatele o vybrané množině dat, nastavení analýzy

---

8 Mgr. Matej Vrtich, e-mailové sdělení 10. 5. 2021

9 „Since different operating systems have different path name conventions, there are several versions of this module in the standard library. The `os.path` module is always the path module suitable for the operating system Python is running on, and therefore usable for local paths.“

a interpretaci výsledků. Všechny výše uvedené vstupní i výstupní parametry musí být nejpozději před publikací definovány ve vlastnostech nástroje; jejich přehled lze nalézt v podkapitole 5.2.1.

### 5.1.2 Připojení databáze

Pro připojení dendrochronologické databáze bylo v ArcGIS Pro nejprve založeno nové databázové připojení využívající platformu PostgreSQL a zadány všechny potřebné údaje, včetně uživatelského jména a hesla, které byly do připojení uloženy. Tímto krokem vznikl ve složce projektu soubor připojení SDE (Spatial Database Engine), umožňující přístup do databáze. Soubor připojení obsahuje všechny údaje potřebné pro přístup k databázi, a pokud je v něm uloženo i přihlašovací jméno a šifrované heslo, nemusí být zadávány při každém požadavku na tuto databázi (ESRI, 2021o). Je potřeba, aby byl tento soubor následně pomocí webového Server Manageru registrován do interního registru serveru, oproti kterému ArcGIS Server ověřuje data projektu před tím, než proběhne publikace nástroje jako webové služby. To je důležité pro to, aby data, na která je odkazováno, nebyla místo pouhého odkazu na jejich umístění celá zkopírována na server na místo, které si určí sám a uživatel se k němu nedostane<sup>10</sup>.

Alternativním přístupem pro tvorbu připojení k databázi by mohlo být například vytvoření nového souboru připojení (connection file) při každém volání spuštění skriptu. K tomu může posloužit funkce `CreateDatabaseConnection`, kdy by byl soubor připojení bez uložených přístupových údajů při každém běhu znovu vytvořen v nově vygenerovaném umístění (pomocí modulu `tempfile` a funkce „`mkdtemp`“) a na konci skriptu by bylo toto umístění opět vymazáno (pomocí modulu `shutil` a funkce „`rmtree`“) (McIntosh, 2018). Upravený kód tohoto přístupu je uveden ve výpisu kódu Výpis 1.

```
sdeTempPath = tempfile.mkdtemp()
arcpy.management.CreateDatabaseConnection(
    out_folder_path=sdeTempPath,
    out_name="DendroConnect.sde",
    database_platform="POSTGRESQL",
    instance="datamap.natur.cuni.cz,5434",
    account_authentication="DATABASE_AUTH",
    username="*****",
    password="*****",
    save_user_pass="DO_NOT_SAVE_USERNAME",
    database="dendrochrone")
<další kód nástroje>
shutil.rmtree(sdeTempPath)
```

*Výpis 1: Alternativa tvorby SDE connection file  
(Zdroj: ESRI, 2021p)*

---

<sup>10</sup> Mgr. Matej Vrtich, e-mailové sdělení 13. 7. 2021

### 5.1.3 Výběr dat

Pro přístup k dendrochronologické databázi byla použita funkce MakeQueryLayer, do níž byl kromě jména výstupního TableView vložen odkaz na connection file (soubor připojení) s příponou SDE. Zároveň byl zadán jakožto formátovací řetězec i filtrovací dotaz ve formátu SQL, formulovaný v závislosti na struktuře databáze H.-M. Majerem (Majer, 2021), přizpůsobený potřebám skriptu. Do něj byly dosazeny proměnné obsahující uživatelsky navolené filtrovací parametry. Tento požadavek na databázi byl definován dvakrát, z toho poprvé je parametrem pouze výběr datasetu surových nebo detrendovaných dat, a předmětem jsou tudíž všechna data zvoleného datasetu; cílem je zjištění celého rozsahu dostupného datasetu. Druhé volání funkce obsahuje kromě totožného výběru datové sady i ostatní filtrovací parametry udávající požadované charakteristiky vzorkovaných stromů; jsou to:

- minimální délka chronologie
- minimální výška stromu
- maximální výška stromu
- minimální výška nasazení koruny stromu
- maximální výška nasazení koruny stromu
- minimální průměr stromu ve výšce hrudi
- maximální průměr stromu ve výšce hrudi.

Průměr stromu ve výšce hrudi (DHB, diameter at breast height) neboli výčetní tloušťka kmene se měří v tzv. výčetní výšce, která by se měla co nejvíce blížit 130 cm od paty kmene. Jsou ale definovaná určitá specifika metodiky sběru tohoto údaje, v závislosti na sklonu svahu, deformaci či poškození měřeného kmene apod. Obvykle je průměr udáván v milimetrech (Metodika venkovního sběru dat, b.r.), databáze však obsahuje hodnoty v centimetrech.

Pro výběr podle zadaného rozsahu parametrů byl zvolen otevřený interval, takže případné záznamy přesně odpovídající zvoleným mezním hodnotám jsou vždy součástí výběru. Zároveň nejsou ve vybrané množině záznamů záznamy s případnou neuvedenou (null) hodnotou výběrových parametrů.

Po získání dat z databáze je tabulka typu TableView za účelem přiřazení identifikátoru „OBJECTID“ jednotlivým záznamům uložena do umístění scratchFolder jako TXT soubor. Při původním postupu kopírování TableView s daty získanými z databáze do geodatabáze scratchGDB se vyskytl problém, kdy kromě přidání pole s OBJECTID navíc ještě vždy zmizelo

pole „year“. Pravděpodobným důvodem samostatného odebrání tohoto pole bylo to, že tento název kolidoval s názvy rezervovanými pro specifické potřeby geodatabáze, což je množina vyhrazených slov, kterým je potřeba se při definování názvů vyhnout. Kdyby šlo o některé méně zásadní pole tabulky, v případě kopírování celého obsahu tabulky by nemusela být jeho absence ani odhalena. Protože není možné ze skriptu zasahovat do názvů polí dendrochronologické databáze a nebyla tak možná editace existujícího názvu, byl tento problém nakonec obejit právě pomocí ukládání do scratch složky ve formátu TXT, které rovněž přiřadí do tabulky pole s OBJECTID, ale žádné jiné neodstraní. Poté jsou data zkopírována zpět do tabulky typu TableView.

Následně je využita funkce SearchCursor z Data Access modulu, která vytváří objekt kurzor, který v rámci cyklu postupuje po řádcích výběru z databáze a umožňuje přístup k jejich obsahu. Při volání funkce je ale potřeba zadat názvy polí tabulky, ke kterým má přistupovat, a pouze tato pole budou v cyklu dostupná. Podle pořadí v zadaném seznamu jmen polí jsou těmto jménům přiřazeny indexy, a pod odpovídajícím indexem lze přistupovat k obsahu daného pole v každé řádce, kterou kurzor projde. Při použití „with“ syntaxe není potřeba odstraňovat kurzor po ukončení používání, bez ní ale ano. Zmíněné principy fungování funkce platí i u ostatních kurzorových funkcí (ESRI, 2021q).

Tímto způsobem jsou tedy z tabulky celého datasetu získány seznamy kódů existujících stanovišť a vzorků. Z výběrové tabulky je získán seznam kódů stanovišť, který je následně porovnán se seznamem všech stanovišť, čímž jsou odvozeny množiny a počty stanovišť, která se do výběru dostala/nedostala. Při porovnání seznamů s kódy je využito setu, protože pro set je potenciálně potřebný čas k provedení operace „x in s“ nezávislý na počtu prvků<sup>11</sup> (PythonWiki, 2020; Do Prado, 2019). Později je rovněž odvozen počet vzorků ve výběru, který lze porovnat s celkovým počtem vzorků. Dále je pak z tabulky výběru získán seznam, ve kterém je pro každé stanoviště z výběru sestaven „balíček“ základních údajů, obsahující kód stanoviště, název stanoviště, druh měřených stromů, zeměpisnou šířku, zeměpisnou délku a také další dvojici zeměpisné délky a šířky v tomto pořadí, pro budoucí specifikaci geometrie. Použití takto seskupených údajů je popsáno podrobně v podkapitole 5.1.5.

Hlavním údajem, který je z databáze čerpán, rovněž pomocí funkce SearchCursor, jsou letokruhové přírůstky. Jejich hodnoty jsou sesbírány po stanovištích a v rámci stanovišť po

---

11 Ing. Vojtěch Svoboda, PhD., e-mailové sdělení 9. 6. 2021

jednotlivých vzorcích. Pomocí seznamu kódů lokalit vyskytujících se ve výběru je pak ze shromážděných dat vytvořen slovník obsahující kódy lokalit jako klíče; pod každým klíčem je uložena hodnota, kterou je podslovník obsahující identifikátory vzorků („sample\_id“) jako klíče. Každý takovýto klíč vzorku má jako svou hodnotu další podslovník, obsahující dva klíče: „years“ a „values“. Pod klíčem „years“ se vždy nachází kontinuální seznam let, ve kterých je znám letokruhový přírůstek daného stromu; pod klíčem „values“ je potom seznam těchto hodnot velikostí přírůstků, kde každý koresponduje svým indexem s indexem jemu odpovídajícího roku (na každý rok připadá jedna hodnota přírůstku).

Takto sestavený „nadslovník“ má takovou strukturu, aby postupně každá v něm se vyskytující lokalita mohla vstupovat do speciální funkce počítající C-hodnoty.

#### 5.1.4 Výpočet Cropperových hodnot

Pro výpočet C-hodnot byla vytvořena zvláštní funkce. Do této funkce vstupují data vždy za jedno stanoviště, a to jako hodnota položky slovníku všech stanovišť pod klíčem odpovídajícím vždy kódu daného stanoviště. Touto hodnotou položky je slovník obsahující identifikátory jednotlivých vzorků jako klíče. Struktura vstupu do funkce je tedy následující (parametry označeny lomenými závorkami):

```
{„<identifikátor vzorku A>“:
    {„years“: [⟨seznam let, pro která jsou k dispozici data⟩],
    „values“: [⟨seznam přírůstků v těchto letech⟩] },
„<identifikátor vzorku B>“:
    {„years“: [⟨seznam let, pro která jsou k dispozici data⟩],
    „values“: [⟨seznam přírůstků v těchto letech⟩] }
(...)}
```

Pro velikost časového plovoucího okna byl zvolen rozsah 13 let, tedy současný rok a 6 let na každou stranu, což jsou v dnešní době typicky upřednostňované hodnoty (Jetschke, van der Maaten a van der Maaten-Theunissen, 2019). Nejprve je tedy zjištěn rozsah let, pro které je možné vypočítat C-hodnotu pro první ze vzorků lokality. Tento rozsah odpovídá seznamu všech let s měřenými přírůstky vzorkovaného stromu, ze kterého je ovšem odstraněno prvních a posledních 6 let, tedy vždy dosah jedné strany plovoucího okna, protože v těchto letech by nebylo dostatečné množství okolních hodnot, ze kterých C-hodnotu vypočítat.



Následně je pro každý pro výpočet použitelný rok tato hodnota spočítána podle vzorce (Obrázek 4). Pro každý střední rok plovoucího okna („mid\_year“) je spočítán obecný (nevážený) klouzavý průměr z hodnot let mid\_year-6 až mid\_year+6, tedy „speciální případ váženého průměru, kdy váhy jsou všechny stejné“ (Ježek, 2015, s.61). Dále je vypočítána druhá mocnina součtu odchylek hodnot jednotlivých roků okna od průměru hodnot okna. Směrodatná odchylka je pak zjištěna vydělením hodnoty z předchozího kroku dvojnásobkem jedné strany okna (tedy 2×6 let, což odpovídá N-1, kde N je počet měření, v tomto případě let v rámci okna) a následným odmocněním.

$$C_t := \frac{X_t - \bar{X}_t^{(n)}}{sd_t^{(n)}} \quad \text{with} \quad \bar{X}_t^{(n)} = \frac{1}{2n+1} \sum_{k=-n}^n X_{t+k} \quad \text{and} \quad sd_t^{(n)} \\ = \sqrt{\frac{1}{2n} \sum_{k=-n}^n \left( X_{t+k} - \bar{X}_t^{(n)} \right)^2}.$$

Obrázek 4: Vzorec pro výpočet Cropperovy hodnoty  
 „t“ značí rok, ve kterém je hodnota počítána, „n“ velikost jedné strany pohyblivého okna  
 (Zdroj: Jetschke, van der Maaten a van der Maaten-Theunissen, 2019)

Posledním a hlavním výpočtem je vydělení hodnoty přírůstku ve středním roce okna s odečteným průměrem okna vypočítanou směrodatnou odchylkou. Tento výsledek je následně zaokrouhlen a přidán do slovníku C-hodnot daného vzorku pod klíč odpovídající střednímu roku okna. Tento výpočet proběhne pro každý vzorek v rámci daného stanoviště. Pak jsou vypočítané C-hodnoty pro každý vzorek daného stanoviště pro všechny roky, ve kterých bylo možné je zjistit, zařazeny do uspořádaného slovníku, ve kterém jsou hodnoty seřazeny podle vzrůstajících let, a následně a zároveň jsou z něj eliminovány veškeré roky, ve kterých jsou k dispozici C-hodnoty z méně než 5 vzorků (Mík, 2020). Tento limit byl zvolen arbitrárně.

Pro všechny roky v rámci lokality, ve kterých je vypočítáno 5 a více C-hodnot, je následně vypočítán medián z těchto hodnot (a zaokrouhlen na 2 desetinná místa místa), a ten je přiřazen do výsledného uspořádaného slovníku pod odpovídající rok. Výstupem z funkce je tedy slovník obsahující pouze jednu sadu klíču, totiž let, a jejich hodnot – mediánů C-hodnot, které patří pod stanoviště, na které byla funkce zavolána. Při těchto výpočtech je pracováno s předpokladem, že žádná chronologie neobsahuje chybějící hodnoty uprostřed (předpoklad spojitosti datové řady).

Funkce pro výpočet C-hodnot je postupně aplikována na všechna stanoviště, jejichž vzorky odpovídaly uživatelským filtrům.

### 5.1.5 Tvorba nové třídy prvků

Pro to, aby mohla proběhnout regresní analýza, bylo zapotřebí nechat vytvořit třídu prvků obsahující v atributové tabulce hodnoty vysvětlované a vysvětlujících proměnných. Pomocí funkce `CreateFeatureClass` je tedy vytvořena bodová třída prvků („sites\_empty“), pro kterou je přes objekt `SpatialReference` nastaven používaný geografický souřadnicový systém `GCS_WGS_1984`, zadáním odpovídajícího EPSG kódu (4326) (ESRI, 2021r; ESRI, 2021s). Pole, která mají být ve výsledné tabulce, se liší tím, jak jsou přidána. Některá je potřeba nejprve vytvořit pomocí nástroje `AddField`, a až pak je vyplnit daty – v tom případě je nutné při jejich tvorbě nastavit jejich typ („TEXT“, „FLOAT“, „SHORT“ apod.) a volitelně definovat další vlastnosti, jako např. počet uchovávaných desetinných míst (platí jen pro formáty desetinných čísel) (ESRI, 2021t). Jiná pole se vytvoří automaticky přímo při funkci přidávající jejich hodnoty.

Do vytvořené třídy prvků jsou tedy nejprve přidána pole pro vybrané údaje o stanovištích z databáze a pro některé hodnoty potřebné pro analýzu. Nejdříve jsou doplněna hodnotami získanými z databáze následující pole:

- „site\_code“ - obsahuje identifikační kód stanoviště
- „site\_name“ - obsahuje název stanoviště
- „species“ - obsahuje druh vzorkovaného stromu
- „latitude“ - obsahuje zeměpisnou šířku ve stupních (přesnost ponechána podle databáze)
- „longitude“ - obsahuje zeměpisnou délku ve stupních (přesnost ponechána podle databáze)
- také byla specifikována geometrie třídy prvků pomocí argumentu „SHAPE@XY“, což je token sloužící k „přečtení hodnot dvojic souřadnic každého prvku“<sup>12</sup> (van Rees, 2021). Souřadnice jsou přiřazeny stejným způsobem, jako u ostatních polí, ale v tomto případě jde o dvojice souřadnic x a y sdružených do n-tice, tedy např. (14.4324466,50.5831543) - nejprve zeměpisná délka, následně zeměpisná šířka (ESRI, 2021u; Lysák, 2021).

Všechny výše uvedené údaje jsou vkládány pomocí `InsertCursor`, což je funkce generující objekt kurzor, který postupně vytváří jednotlivé záznamy tabulky - pomocí metody `insertRow` vkládá řádky vyplněné hodnotami z předchystaného slovníku. Hodnoty přiřazené v rámci argumentu

12 „reading the coordinate value pairs for each feature“

„SHAPE@XY“ nevytvoří sloupec ve výsledné tabulce, ale jak bylo již zmíněno výše, specifikují geometrii třídy prvků (ESRI, 2021v).

Poté je využita funkce `ExtractMultiValuesToPoints` pro extrahování hodnot rasterových podkladů v místech měřených lokalit podle jejich souřadnic. Podle výchozího nastavení je přiřazena vždy hodnota středu buňky, do které spadá umístění dané lokality. Tato funkce automaticky přidává pole se jménem specifikovaným při volání funkce, odpovídajícím názvu vysvětlující proměnné, kterou raster znázorňuje. Rastery (viz podkapitola 4.2.2) jsou sdíleny na serveru, a proto je na ně odkazováno pomocí odkazu na server přes REST rozhraní. Jsou přidána tři pole, která budou k dispozici pro použití jako vysvětlující proměnné v rámci regresní funkce:

- „elevation“ - hodnoty nadmořské výšky
- „slope“ - hodnoty sklonu terénu
- „aspect“ - hodnoty reklasifikované orientace plochy terénu.

Zde je potřeba říci, že kdyby dendrochronologická databáze disponovala kompletní sadou údajů o všech třech výše zmíněných charakteristikách, nebylo by potřeba tyto hodnoty získávat v průběhu z rasterových podkladů a stačilo by je shromáždit z ní.

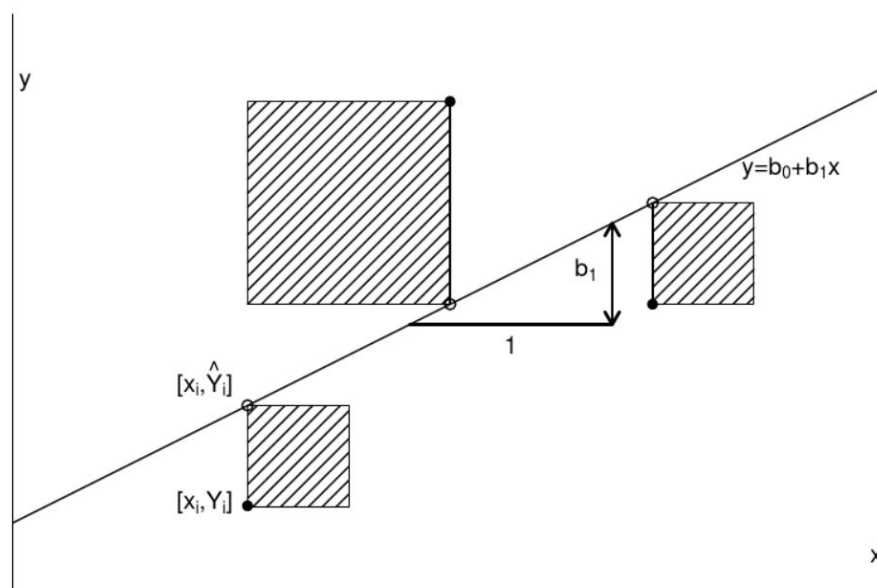
Zbývající hodnoty jsou doplněny do atributové tabulky třídy prvků „sites\_empty“ z již dříve vytvořených slovníků pomocí `UpdateCursor`, což je opět funkce vytvářející objekt kurzor, který se posunuje po jednotlivých záznamech tabulky a v závislosti na kódu stanoviště doplňuje do předchystaných polí ze slovníku mediánů odpovídající hodnotu mediánu C-hodnot a ze slovníku počtů vzorků odpovídající číslo. Po doplnění každé řádky novými daty je zavolána metoda `updateRow`, která vždy aktualizuje současnou řádku (ESRI, 2021w; ESRI, 2021x). Pole naplněná daty funkcí `UpdateCursor` jsou tedy následující:

- „Cm<zvolený rok>“ - obsahuje medián C-hodnot na daném stanovišti ve zvoleném roce (v názvu pole je doplněn rok, ve kterém má dle volby uživatele proběhnout analýza)
- „samples\_n“ - obsahuje počet vzorků, které byly zahrnuty do výpočtu mediánu C-hodnot na daném stanovišti; zařazeno kvůli zpětnému ověření robustnosti výpočtu v tomto stanovišti.

### 5.1.6 Regresní funkce

Pro aproximaci mediánů C-hodnot v oblasti vymezené pro tuto případovou studii byla použita obecná lineární regrese (Generalized Linear Regression). Podle dokumentace ArcGIS Pro tato

funkce slouží k predikci nebo modelování závislé veličiny pomocí zadaných vysvětlujících proměnných. Umožňuje regresi spojitých jevů pomocí zobecněné metody nejmenších čtverců – ordinary least squares (OLS) (ESRI, 2021y). Tato metoda je založena na minimalizování sumy druhých mocnin odchylek aproximovaných hodnot od původních hodnot při prokládání regresní funkce (Ježek, 2015). Princip metody je patrný ze schematického znázornění proložení přímky na Obrázku 5.



Obrázek 5: Princip metody nejmenších čtverců  
(Zdroj: Zvára, 2013)

Vstupem do regresní funkce je vytvořená třída prvků „sites\_empty“ ze scratchGDB s doplněnými vypočítanými mediány C-hodnot a dalšími potřebnými daty, dále pak bodová vrstva ze serveru (Hosted Feature Layer) s lokalitami pro aproximaci, na kterou je odkazováno odkazem na server přes REST rozhraní. Jako typ modelu byl zvolen „CONTINUOUS“ (souvislý, Gaussův), protože se očekává, že vysvětlovaná proměnná je spojitý jev. Použitým druhem regrese je tedy metoda nejmenších čtverců (ESRI, 2021y). Vysvětlující proměnné jsou vloženy jako seznam proměnných zadaných uživatelem. Při vytvoření a naplnění bodové vrstvy byla použity záměrně stejné názvy polí se stejným typem obsahu jako v databázi, aby později v regresní funkci mohly být přiřazeny k sobě. V případě, že by názvy odpovídajících polí v databázi a v bodové vrstvě nebyly stejné, bylo by nutné vytvořit seznam seznamů obsahujících dvojice názvů polí, které vyjadřují stejný jev (ve správném pořadí), a použít je jako parametr regrese „explanatory\_variables\_to\_match“. V opačném případě ale tento krok není nutný. Výstupy z regresní funkce se ukládají do scratchGDB, a jsou to:

- „`regression_output`“ – bodová vrstva vycházející z umístění vstupních lokalit, obsahující vypočítané mediány C-hodnot, vysvětlující proměnné, vysvětlované hodnoty mediánů C-hodnot, reziduály (odchylky predikovaného od pozorovaného), standardizované reziduály, a také grafy charakterizující vztahy mezi jednotlivými proměnnými apod.
- „`regression_output_predicted`“ – bodová vrstva, umístění jejíchž bodů odpovídá síti bodů ve vstupní vrstvě. Obsahuje vysvětlující proměnné a predikované mediány C-hodnot. Je východiskem pro další analýzu.

### 5.1.7 Interpolační funkce

Pomocí regrese vytvořená vrstva „`regression_output_predicted`“ potom vstupuje do interpolační funkce (Local Polynomial Interpolation). Tato funkce byla zvolena na základě toho, že je očekávána variabilita sledované veličiny na lokální úrovni, tedy jde o změny na kratších vzdálenostech (ESRI, 2021z). Velikost buňky byla nastavena na  $0.0009^\circ$ , což přibližně odpovídá vzdálenosti 100 metrů (USNA, 2019). Stupeň polynomu odpovídá zvolenému vstupu od uživatele, od 0 do 3. Důležitým parametrem je také vymezení toho, které body se známou hodnotou v okolí predikovaného bodu mají být zahrnuty do jeho výpočtu. Pro tyto účely slouží nastavení tzv. `search neighborhood`. Předpokladem tohoto interpolačního výpočtu je totiž existence poklesu prostorové autokorelace (tedy závislosti výskytu sledovaného jevu v prostoru na výskytu téhož jevu v blízkém okolí (Cliff a Ord, 1973)) sledované veličiny s rostoucí vzdáleností od místa predikce.

Příliš vzdálené body mají nejen zcela zanedbatelný nebo žádný vliv, nýbrž mohou být i negativním či zkreslujícím faktorem pro interpolační výpočet, v závislosti na míře rozdílnosti oblastí, ve kterých se tento vzdálený bod a interpolovaný bod nachází (ESRI, 2021aa). Proto se používá prostorové omezení množiny bodů, u kterých se očekává vliv, právě oblastí definovanou parametrem `search neighborhood`. Pro oblast `search neighborhood`, pro kterou byl vybrán kruhový tvar (viz níže), byl zvolen poloměr  $0.0045^\circ$ , což přibližně odpovídá 500 metrům, tedy pětinasobku velikosti strany pixelu výstupního rasteru (pětinasobek zvolen podle Křikavová, 2009) a také kolmých vzdáleností aproximovaných lokalit; množství započítaných bodů nebylo početně omezeno. Jediným omezením je výchozí omezení počtu zahrnutých sousedních bodů podle zvolené interpolační metody, kterým je pro lokální polynomičtí interpolaci hranice 1000 bodů. Vyhledávací okno `search neighborhood` nebylo nijak natočeno ani děleno do sektorů,

protože body jsou rozmístěny rovnoměrně, a také proto, že se nepředpokládá konkrétní směrové ovlivnění (ESRI, 2021aa). Bylo použito Standardní kruhové search neighborhood (Standard Circular Search Neighborhood), protože není předpokládána konkrétní směrová prostorová anizotropie aproximované veličiny, takže by její změna neměla záviset na směru (Ježek, 2015). Pro výpočet predikce z interpolační oblasti byla zvolena Gaussova funkce (kernel\_function = „GAUSSIAN“), která směrem ke kraji search neighborhood klesá rychle k kladnému/zápornému nekonečnu a bližší sousední body tak mají výrazně vyšší váhu, naproti tomu zcela okrajové body vliv nemají (ESRI, 2021z). Výstupem z interpolační funkce je plocha predikovaných C-hodnot v oblasti severu Kokořínska (v rozsahu vstupní bodové vrstvy), výsledný raster je uložen do scratchFolder pod názvem „interpol\_r“.

Původně probíhalo testování i pro celou Českou republiku, ale pravděpodobně i kvůli velmi malému množství dat docházelo v současné verzi při interpolaci na takto velké ploše při použití zeměpisné šířky a délky jako vysvětlujících proměnných k výsledkům, které odpovídaly převážně pouze obecným severo-j jižním, resp., západo-východním gradientům bez lokální variability (vyjma oblasti Kokořínska mezi měřenými lokalitami). Zároveň vycházely predikované hodnoty na většině území nesmyslně extrémní. Mezi hodnotami lokalit existuje nejspíše slabý směrový trend, který se s rostoucí vzdáleností od měřených lokalit amplifikuje, a proto by regresní analýza pro území větších rozměrů (např. celé CHKO) by za současného množství vstupních hodnot nebyla smysluplná. Lze ale očekávat, že při větším množství měřených lokalit, rovnoměrněji rozmístěných po interpolační ploše a rovněž v lepším případě i zachycujících co nejvíce různých charakterů krajiny (podle volených vysvětlujících proměnných) budou výsledky analýzy lépe aproximovat hodnoty pro větší území i o velikosti České republiky. Dále by bylo možné využít přepočítání vybraných vysvětlujících proměnných pomocí vah, vycházejících z míry vlivu daného faktoru na letokruhové přírůstky.

Obecně ale regresní analýza a interpolace pro celou Českou republiku za využití mřížky bodů o rozestupech 100 metrů trvala příliš dlouho, a proto bylo přistoupeno pouze k případové studii pro oblast severu Kokořínska, odkud pochází měřená data. Pro kratší trvání by musely být zvětšeny bodové rozestupy, čímž by ale docházelo k zanedbávání lokální variability vysvětlujících proměnných, a tedy i ke zkreslení výsledků. Dále by mohlo zrychlit proces zmenšení poloměru search neighborhood či omezit v něm počet započítaných bodů (ESRI, 2021aa). Nebo v případě, že by nebylo nutné získat výsledky pro celou ČR, postačovalo by použít pro dopočítání hodnot regresí i bodovou vrstvu oříznutou vlastním menším polygonem –

v kontextu aplikace by bylo možné vytvořit sadu bodových vrstev, oříznutých podle menších správních jednotek (kraje, okresy), či spíše přirozeněji podle hranic krajinných celků, chráněných oblastí apod. Rovněž by bylo možné využít propojení widgetu s funkcemi některého existujícího widgetu Web AppBuilderu, určenými pro kreslení, čímž by bylo sice o něco méně exaktně, ale zato zcela libovolně a přímo v aplikaci, uživateli umožněno vytvoření vlastního polygonu pro interpolaci, kterým by pak stačilo oříznout bodovou vrstvu naplněnou daty potřebnými pro analýzu, a pak by bylo možné použít ho jako omezení interpolační oblasti (ESRI, 2021ab).

Po proběhnutí interpolace je funkcí ExtractMultiValuesToPoints přidáno do atributové tabulky nové pole obsahující predikované hodnoty z interpolovaného rasteru, které odpovídají prostorovému umístění měřených stanovišť; tyto hodnoty jsou následně zaokrouhleny díky nástroji CalculateField na dvě desetinná místa, stejně jako původní vypočítané hodnoty. Dále je vytvořeno další nové pole, do kterého je následně opět s využitím nástroje CalculateField vypočítán rozdíl mezi predikovanou hodnotou mediánu C-hodnot v dané lokalitě, a původní hodnotou, vypočítanou z naměřených dat; tyto odchylky jsou rovněž zaokrouhleny (na 3 des. místa). V obou případech přepočtů je využit typ výrazu „PYTHON3“, ve kterém musí být názvy polí obklopeny vykřičníky (např. round(!{Cm\_fieldname}!,3)-!{pred\_name}!). Přidaná pole jsou tedy „pred<zvolený rok>“ obsahující predikce pro místa měřených lokalit a „dev<zvolený rok>“ obsahující odchylky predikcí od vypočítaných hodnot lokalit (ESRI, 2021ac;ESRI, 2021ad).

### **5.1.8 Reklasifikace**

Kvůli existenci chyby v rámci mapové služby Esri, která v době návrhu této webové aplikace nedovolovala mapové službě vytvořené programem ArcGIS Pro vykreslovat rasterové vrstvy, bylo přistoupeno k uzpůsobení prezentace výsledků analýzy uživateli. Výsledná rasterová vrstva má být v budoucnu dostupná po proběhnutí geoprocessingu ke stažení, zatímco pro účely zobrazení výsledných interpolovaných dat v aplikaci bylo využito funkce Reclassify pro překlasifikování hodnot do intervalů pomocí objektu RemapRange, a následně funkce Raster to Polygon. Tato funkce převede vždy stejné hodnoty reklasifikovaného rasteru na jeden vícedílný (multipart) polygon, čímž je získána vektorová polygonová vrstva, obsahující takový počet prvků, kolik je nastaveno nových tříd při reklasifikaci. Multipart polygony byly zvoleny z toho důvodu, že pro účely této analýzy není potřeba získat jednotlivé polygony, ale spíše jednotlivé třídy, resp. pro každé místo v oblasti interpolace konkrétní třídu, která je pro toto místo

aproximována. V případě této aplikace se jedná až o 13 tříd; hranice jejich intervalů jsou nastaveny pomocí vlastní funkce v rámci skriptu, která je sestaví podle zadaného kroku a částečně dynamicky podle vstupního rasteru interpolovaných hodnot.

Do funkce vstupuje minimum a maximum tohoto rasteru. Nejprve je zvolena z absolutní hodnoty minima a absolutní hodnoty maxima rasterového výsledku interpolace ta vyšší („outer“), která je následně zaokrouhlena nahoru na dvě desetinná místa. Pak jsou podle kroku (step), který byl pro tuto případovou studii zvolen na 0.15, vygenerovány třídy. Je vytvořena jedna nulová třída, obsahující pouze nulové hodnoty, a je překlasifikována rovněž na hodnotu 0. Třída nulových hodnot je ve skutečnosti nastavena v rozsahu hodnot -0.001 až 0.001, kvůli polootevřenosti intervalů navazujících hraničními hodnotami. Vzhledem k tomu, že je zde pracováno s hodnotami zaokrouhlenými na dvě desetinná místa, obsahuje ale množina vstupujících rasterových hodnot překlasifikovaných na hodnotu 0 právě pouze hodnotu 0.00.

Pro ostatní třídy platí rozdělení od záporných hodnot po kladné tak, že každá horní mez třídy do této třídy patří, dolní mez pak nikoliv. Výjimkou je třída s nejnižšími hodnotami, do které patří i její dolní mez (ESRI, 2021ae). Meze dalších tříd 1 až 5, resp. -1 až -5, jsou stanoveny jako kladné, resp. záporné násobky kroku. V případě, že existují takové hodnoty výsledného rasteru, které převyšují vnější meze tříd s hodnotami 5 nebo -5, pak jsou všechny tyto přesahující hodnoty zařazeny do přepadové třídy 6 nebo -6 v závislosti na jejich kladnosti nebo zápornosti. Vnější mez třídy 6/-6 se vždy rovná nejvyšší absolutní hodnotě rasteru, se znaménkem odpovídajícím znaménku třídy. Přerozdělení hodnot rasteru do nových tříd je vypsáno v Tabulce 3, stručněji pak v Tabulce 4.



Od původní hodnoty rasteru		Do původní hodnoty rasteru		Reklasifikovaná hodnota
Hodnota	V kódu	Hodnota	V kódu	
-max. absolutní hodnota rasteru	-outer	-0.75	-5*step (= -border)	-6
-0.75	-5*step (= -border)	-0.60	-4*step	-5
-0.60	-4*step	-0.45	-3*step	-4
-0.45	-3*step	-0.30	-2*step	-3
-0.30	-2*step	-0.15	-1*step	-2
-0.15	-1*step	-0.001	-0.001	-1
-0.001	-0.001	0.001	0.001	0
0.001	0.001	0.15	1*step	1
0.15	1*step	0.30	2*step	2
0.30	2*step	0.45	3*step	3
0.45	3*step	0.60	4*step	4
0.60	4*step	0.75	5*step	5
0.75	5*step (= border)	max. absolutní hodnota rasteru	outer	6

*Tabulka 3: Reklasifikace rasteru s interpolovanými hodnotami mediánů Cropperových hodnot (Zdroj: autorka)*

Původní hodnoty rasteru	Reklasifikovaná hodnota
-outer až -border	-6
-border až 0 (vyjma 0), rozděleno do 5 tříd	-5 až -1
0	0
0 až border (vyjma 0), rozděleno do 5 tříd	1 až 5
border až outer	6

*Tabulka 4: Stručný přehled reklasifikace rasteru (Zdroj: autorka)*

Poznámka: „outer“ a „border“ jsou absolutní hodnoty maxima a minima rasteru; „outer“ odpovídá vyšší z nich, „border“ té nižší.

Toto použité řešení má nevýhodu v tom, že při velmi malém rozsahu hodnot blízko kolem 0 se může stát, že všechny hodnoty budou spadat do velmi malého množství tříd (např. pouze do jedné), nebo naopak bude rozsah hodnot velmi široký a jejich převážná část bude patřit do dvou okrajových tříd. V obou těchto případech budou v reklasifikovaném výstupu hodnoty rozřazeny

do tříd velmi nerovnoměrně. Avšak hlavní výhodou, a také důvodem, proč byl tento způsob reklasifikace nakonec zvolen jako lépe vyhovující, ale je to, že díky stejně velkému kroku, platícímu pro všechny skriptem vygenerované výstupy, je pak možné tyto výstupy vzájemně porovnávat v rámci stejné legendy. Také by bylo možné při rozšíření aplikace nastavit při každém spuštění uživatelem zvolený krok jako jeden z parametrů, ale současný stav skriptu tuto možnost neimplementuje. Rovněž by pevný krok neměl vadit proto, že pro C-hodnoty je zásadní zejména otázka jejich kladnosti nebo zápornosti, což tento způsob klasifikace zachovává.

Alternativní možností vytvoření nových tříd výstupu ve webové aplikaci by bylo využití dynamického kroku. V tomto případě by byla např. zvolena z absolutních hodnot minima a maxima rasterového výsledku interpolace ta nižší („border“), a z ní by po zaokrouhlení nahoru byla odvozena velikost kroku intervalu („step“) jejím vydělením číslem 5, a s tímto intervalem by bylo vytvořeno 5 kladných a 5 záporných tříd. Zbývající hodnoty přesahující absolutní hodnotu hranice „border“ by byly přiřazeny do kladné, resp. záporné okrajové třídy (6, resp. -6), jejíž vzdálenější hranice od nuly odpovídá zbývajícím (tedy větším) absolutním hodnotě vybrané z minima a maxima rasteru („outer“). Nulová třída by zůstala stejná. Takto vzniklé třídy by tedy opět popisovaly veškeré existující hodnoty rasteru. Kód pro vytvoření `remapRange` pro tuto alternativu lze nalézt ve Výpisu 2.

Rovněž by bylo možné při tomto přetřídění získat velikost kroku vydělením naopak vyšší z absolutních extrémů – třídy by pak byly širší a tedy o něco méně podobné, ale krok stále založen na rozsahu rasteru; zároveň by odpadla třída 6/-6 (jejíž velikost se neodvíjí od velikosti kroku), a všechny hodnoty by spadaly do tříd od -5 do 5. Při reklasifikaci na základě dynamického kroku by bylo výhodou, že by v případě první alternativy byly vždy využity všechny třídy 1 až 5 (-1 až -5), a dále i jedna ze tříd 6 nebo -6 (ovšem kromě situace, kdy by byla absolutní hodnota minima rasteru rovna absolutní hodnotě maxima rasteru, a žádná 6. třída by nebyla potřeba). Cílem by bylo pro případ, že by měl výstupní raster velký rozsah hodnot, aby byly hodnoty v blízkosti nuly znázorněny podrobněji než odlehlejší hodnoty.

Pro druhou alternativu by zase platila vždy existence 5 tříd na stranu se vzdálenějšími hodnotami od nuly, a maximálně 5 tříd na stranu opačnou. Naopak nevýhodou těchto přístupů s dynamicky odvozeným krokem by ale byla právě nemožnost porovnání výstupních vrstev v aplikaci mezi sebou podle stejné legendy. Z tohoto důvodu bylo přistoupeno k výše zmíněnému řešení s pevně stanoveným krokem. Toto vše se týká pouze reklasifikované

polygonové vrstvy prezentované v aplikaci; výstupní interpolační raster, ze kterého vektorová vrstva vychází a který bude v budoucnu k dispozici ke stažení, zůstává reklasifikací neovlivněn.

```
def range_list(min,max):
    if abs(min) > abs(max):
        border = abs(max)
        outer = abs(min)
    else:
        border = abs(min)
        outer = abs(max)

    outer = ceil(outer*100)/100 # round up to 2 decimal places
    border = ceil(border*100)/100
    step = border/5

    rangeList = []
    # classes from -6 to 6; 0 equals class 0
    classList = [-(round(outer,2)), -(border), -6] # 1st class
    rangeList.append(classList)

    a=-(border)
    c=-5

    for _ in range(11):
        b=a+step
        if c == 0:
            classList=[-0.0001,0.0001,c] # zero class
        elif c == -1:
            classList=[round(a,2), -0.0001,c] # class -1
        elif c == 1:
            a=0
            b=a+step
            classList=[0.0001,round(b,2),c] # class 1
        else:
            classList=[round(a,2), round(b,2),c]

        rangeList.append(classList)
        a+=step
        c+=1

    classList = [border,round(outer,2),6] # last class
    rangeList.append(classList)
    return rangeList,step
```

*Výpis 2: Alternativa tvorby remapRange  
(Zdroj: autorka)*

### 5.1.9 Symbologie

V reklasifikované výstupní třídě prvků, které se nastavení symbologie v rámci této případové studie týká, se mohou nacházet pouze multipart polygony, které spadají do jedné ze 13 stanovených tříd. Postačuje tedy postačuje definovat symbologii pouze pro ně, a od jejich rozdělení se odvíjí. Zvolená symbologie by měla zohledňovat charakteristiky predikované veličiny. Pro C-hodnoty je zásadní charakteristikou otázka jejich kladnosti nebo zápornosti,

protože poukazuje na relativní nárůst, resp. relativní pokles růstu šířky letokruhů<sup>13</sup>. To by mělo být ve výstupní vrstvě znázorněno. Pro tento účel je tedy navrhována barevná škála se středem v nulové třídě, kde na zápornou stranu od nuly stoupá intenzita červené barvy a na kladnou stranu barvy zelené. Symbologie může být nastavena pomocí plných barev stupňovaného dvojitého gradientu, nebo v souvislosti s nízkým počtem tříd mohou být v zájmu vyššího přizpůsobení nastaveny plné barvy symbologie podle unikátních hodnot pole „gridcode“, obsahujícího reklasifikované hodnoty tříd, ale ve stylu kopírujícím styl odstupňovaných barev vygenerovaného gradientu.

Pro účely nastavení symbologie je možné využít funkci `SetParameterSymbology`, která je určena pro definování symbologie výstupu ze skriptového nástroje. Výše zmíněné symbologii dle unikátních hodnot by odpovídalo využití vykreslovacího objektu `Web Map JSON renderer object` pro unikátní hodnoty, protože výstup by měl být prezentován ve webové mapě. Zmíněná funkce potřebuje dva argumenty. Prvním je index, který opět odpovídá identifikačnímu indexu parametru v rámci skriptu a přiřazuje symbologii k výstupu, na který má být aplikována. V případě vyvíjeného skriptu by šlo tedy o polygonovou vrstvu „`interpolated_polygons`“. Druhým argumentem je řetězec ve formátu JSON, který reprezentuje specifikační objekty webové mapy přiřazené k objektu `JSON renderer` (ESRI, 2021af). V rámci `unique value renderer` je definováno pole, na základě kterého je symbologie založena, výchozí symbologie pro případné nezatříděné hodnoty (tento případ by ale ve výstupní třídě vyvíjeného nástroje nastat neměl), a dále pak specifická symbologie pro jednotlivé třídy (ESRI, 2021ag). Všechny tyto symbologie jsou reprezentovány symbolovými objekty specifikace webové mapy, které mají nastavitelné vlastnosti a atributy. Polygony by měly být symbolizovány objektem typu „`esriSFS`“ (Simple Fill Symbol) se stylem „`esriSFSSolid`“ znázorněným vždy jednou barvou nadefinovanou pomocí barevného modelu RGBA, kde jsou určeny rozsahem 0 až 255 úrovně sytosti barev červené, zelené a modré a také průhlednost, kdy 255 znamená úplnou opacitu (ESRI, 2021ah). Okrajové linie polygonů by pak byly symbolizovány objektem typu „`esriSLS`“ (Simple Line Symbol) se stylem „`esriSLSSolid`“ analogickým způsobem (ESRI, 2021ah). Příklad definice symbologie unikátních hodnot jdoucích od červené barvy (třída -6) k zelené (třída 6) s černou okrajovou linií, který je uveden v Příloze 2, by byl přiřazen výstupní polygonové vrstvě příkazem `arcpy.SetParameterSymbology(23,symbology_renderer)`. Kdyby byl výstupní soubor stanovišť třídou prvků, její symbologie by pak mohla být ve skriptu definována pomocí objektu

---

13 Mgr. Milan Mík, e-mailové sdělení 11. 5. 2021

typu „esriSMS“ (Simple Marker Symbol), např. podle Výpisu 3, a po vytvoření třídy přiřazena k výstupu opět funkcí SetParameterSymbology.

```
symbology_points = """{
  "type": "simple",
  "symbol":
  {
    "type": "esriSMS",
    "style": "esriSMSCircle",
    "color": [255,0,0,255],
    "size": 5,
    "angle": 0,
    "xoffset": 0,
    "yoffset": 0,
    "outline":
    {
      "color": [0,0,0,255],
      "width": 1
    }
  },
  "label": "Stanoviště",
  "description": "Lokalita odběru vzorků.",
}"""
symbology_points_renderer = f"JSONRENDERER={symbology_points}"
```

*Výpis 3: Možné definování symbologie bodové třídy prvků  
(Zdroj: podle ESRI 2021ag)*

Využitím formátovacího řetězce a ošetřením speciálních znaků při definování symbologie ve skriptu řetězcem ve formátu JSON (Příloha 2) by také mohly být doplněny popisky tříd přímo v závislosti na reklasifikačních mezích těchto tříd, čímž by mohly pomoci lepšímu porozumění obsahu výstupní třídy prvků.

Poznámka: pro nástroje, které budou mít stálý přístup k souboru vrstvy formátu .lyrx je jednou z možností také symbologii odvodit od této již existující .lyrx vrstvy jejím definováním ve vlastnostech nástroje (viz podkapitola 5.2.1).

### 5.1.10 Ošetření výjimek

V případě, že pro některá stanoviště, která prošla filtry, nebyl vypočítán medián C-hodnot (např. kvůli nízkému počtu chronologií zasahujících do daného roku), jsou tato stanoviště vypsána v rámci varování, ale geoprocessing není přerušeno. Pro to, aby mohla vůbec regresní analýza proběhnout, by bylo z technologického hlediska pro výpočet zapotřebí minimálně  $n$  hodnot, přičemž  $n$  odpovídá počtu vysvětlujících proměnných zvýšenému o 1.

$$n = \text{len}(\text{explanatory}) + 1$$

Tento limit je ale příliš nízký na to, aby model generoval relevantní výsledky vzhledem k dostupným testovacím datům, a proto bylo toto číslo zvýšeno. Kvůli v současné době

nedostatečnému množství naměřených dat pro relevantní analýzu (pouze 20 lokalit) byla pro tuto případovou studii arbitrárně zvolena hranice 10 hodnot lokalit ve zvoleném roce (podle Mík, 2020). Při rozšíření databáze v budoucnosti bude možné upravit tento limit blíže adekvátnímu omezení, úpravou označeného řádku v kódu nástroje (viz Příloha 1). V současné verzi nástroje tedy platí, že pokud je získaných C-hodnot vstupujících do analýzy pro daný rok méně než 10, nástroj ukončí činnost s ohlášením chyby z důvodu nedostatku dat potřebných pro relevantní analýzu.

### 5.1.11 Log

V rámci skriptu se vyskytuje více úrovní výpisů. Pro koncového uživatele nedůležité informace se vypisují jenom v případě terminálu editoru (pomocí příkazu „print“), nebo navíc i při spuštění nástroje v ArcGIS Pro; důležitější zprávy by měly být uživateli zobrazeny v aplikaci. V ArcGIS Pro se rozlišují tři skupiny hlášení podle úrovně jejich závažnosti:

- Informace – obsah má čistě informativní charakter (závažnost: 0)
- Varování – obsahuje upozornění na něco nestandardního, co samo o sobě může, ale nemusí nutně způsobit pád nástroje, nicméně uživatel by si toho měl být vědom (závažnost: 1)
- Chyba – ohlašuje chybu, kvůli které je běh nástroje přerušen (závažnost: 2).

Podle čísla úrovně závažnosti je také možné později omezit, která hlášení se budou uživateli zobrazovat (ESRI, 2021ai).

Nastavení jednotlivých hlášení se provádí pomocí funkcí `AddMessage`, `AddWarning` nebo `AddError`, jejichž argumentem je řetězec tvořící požadovanou zprávu. Varování a Chyba má výhodu v tom, že se (pokud není zvlášť nastaveno jinak) vypíší všude automaticky. Pro zobrazení hlášení typu Informace (zpráva, Message) je v aplikaci potřeba definovat je jako výstupní řetězec. Toho je docíleno využitím příkazu `SetParameterAsText`, kam je vložen text hlášení jako řetězec, stejným způsobem, jakým tomu je u ostatních výstupů (viz podkapitola 5.1.1).

Některé jednotlivé funkce, které skript využívá, generují svůj vlastní informační log, ale pokud vše proběhne bez problému, k uživateli se žádná zpráva nedostane; log konkrétní funkce je nicméně možné získat do proměnné ve skriptu pro další potřeby pomocí funkce `GetMessages`. Tato funkce vrací výpis poslední funkce, která proběhla před ní. Je dokonce možné nastavit i požadovanou závažnost zachycovaných hlášení s tím, že výchozí je závažnost 0, přičemž

v tomto případě jsou zachycena i hlášení všech vyšších úrovní (ESRI, 2021ai). Využití této funkce bylo navrženo s ohledem na to, že v rámci analýzy volaná regresní funkce sama generuje informační log obsahující charakteristiky proběhlé regrese, které mohou být v případě posuzování analýzy užitečné. Proto je tento výpis informací definován jako jeden z textových výstupů nástroje.

Nástroj by měl tedy podle tohoto návrhu vypisovat v případě bezproblémového proběhnutí následující informace: zvolené parametry filtrovaného výběru, zvolené parametry nastavení analýzy, počet stanovišť alespoň částečně zastoupených ve výběru a z kolika celkově, případná zcela odfiltrovaná stanoviště, počet vzorků zahrnutých ve výběru a z kolika celkově, rozsah délek chronologií vzorků, rozpětí let s vypočítanými mediány C-hodnot ve všech zachovaných stanovištích, počet stanovišť s vypočítaným mediánem C-hodnot ve zvoleném roce, stanoviště vstupující do regrese, charakteristiky regresní funkce a interpretaci výstupní reklasifikované polygonové vrstvy.

V předpřipravených widgetech není možné pro použitý režim služby (viz podkapitola 5.2.2) definovat průběžný výpis logu, nicméně je možné nastavit vypsání požadovaných zpráv i na úrovni Informace alespoň po proběhnutí skriptu, čímž může být uživateli předána zpráva o zvolených parametrech, podmínkách analýzy nebo vysvětlení výsledků geoprocessingu alespoň dodatečně.

## **5.2 Publikace geoprocessingové služby**

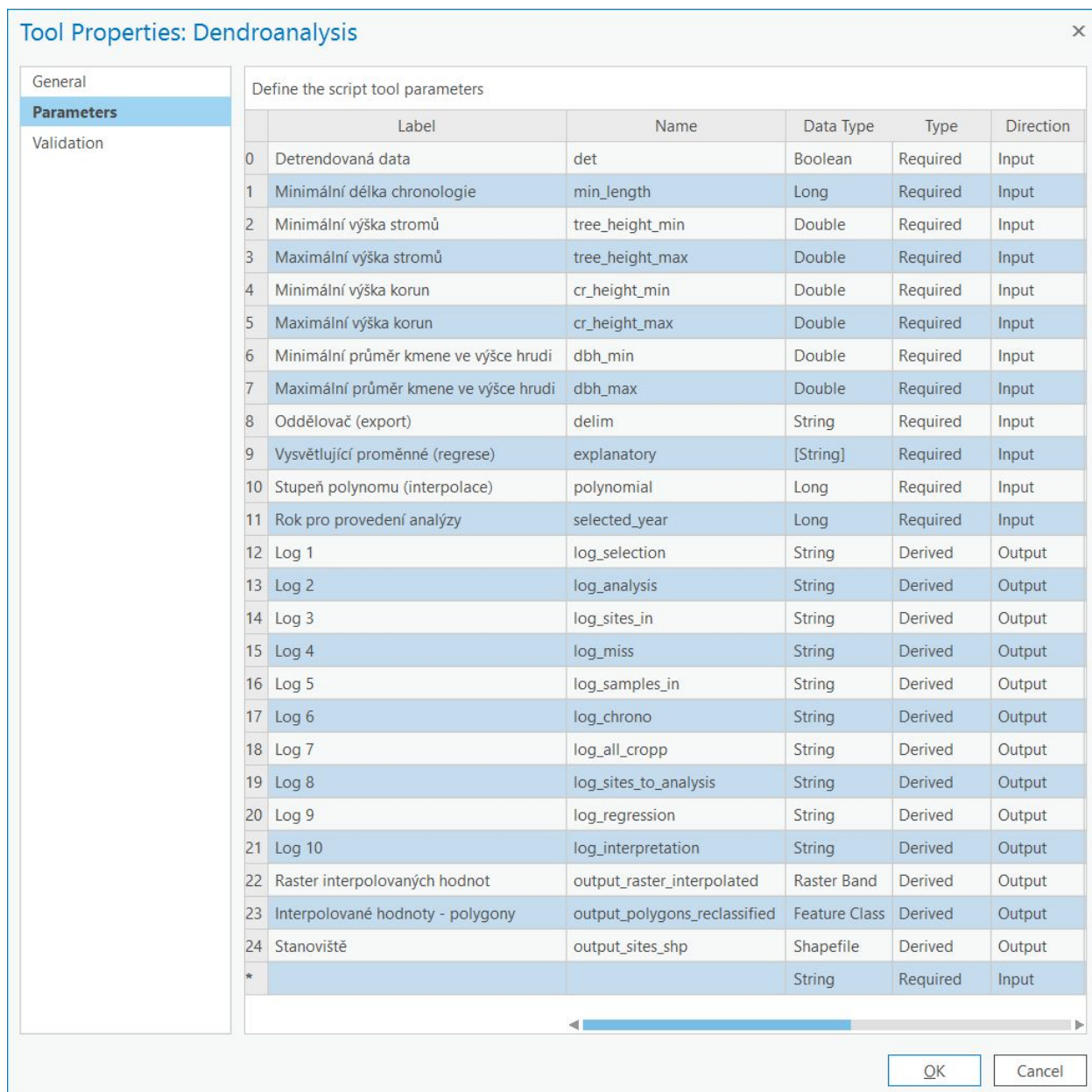
Vytvořený skript byl testován nejprve přes terminál Python editoru VisualStudioCode, poté byl přidán do ArcGIS Pro do Toolboxu projektu jako nový Python skript.

### **5.2.1 Nastavení parametrů nástroje**

Pro každý skript je v ArcGIS Pro potřeba nastavit všechny parametry na stejnojmenné záložce ve vlastnostech skriptu (viz Obrázek 6). Nastavuje se popis a název, dále pak datový typ parametru, zda je vyžadováno jeho zadání, a zda jde o vstupní či výstupní parametr. Datový typ určuje, jaké hodnoty bude uživateli umožněno zadat; jak bude zmíněno v kapitole 5.3, v této fázi nastavený parametr neznámá nutně, že bude muset být v aplikaci zadáván uživatelem. Všechny parametry jsou zadány pod čísla korespondující s jejich indexem ve skriptu. Vstupy

jsou v něm získány pomocí `GetParameterAsText`, výstupy identifikovány díky `SetParameterAsText`. Do výstupů patří výpisy logu, ale zejména výsledky proběhlé analýzy:

- raster interpolovaných mediánů C-hodnot jako pásmo rasteru
- reklasifikovaná polygonová vrstva tohoto rasteru jako třída prvků
- bodová vrstva stanovišť s doplněnými hodnotami jako soubor SHP (Shapefile) (ESRI, 2021aj).



Obrázek 6: Přehled definovaných parametrů  
Zdroj: ArcGIS Pro



## 5.2.2 Nastavení publikace

Aby mohl být nástroj publikován, musí být nejprve spuštěn v ArcGIS Pro a úspěšně proběhnout. Zadané hodnoty (parametry) se stanou po publikaci hodnotami výchozími. Také musí být k ArcGIS Pro připojeno rozhraní Portal for ArcGIS. Poté je potřeba ve vyskakovacím okně proběhnutého GP nebo ve výběru záložek přejít na odkaz „Historie“, kde bude o tom zobrazen záznam. V menu pravého tlačítka se pak zvolí možnost „Sdílet jako...“ a „Sdílet webový nástroj“ (Share web tool), čímž se otevře dialogové okno publikace. Zde je potřeba na záložce „Obecné“ vyplnit podrobnosti včetně klíčových slov. Pro vyvíjený nástroj by měla být zvolena možnost odkazování na registrovaná data, všechna data, na která je odkazováno v nástroji, musí být umístěna a registrována na serveru. Pokud data nejsou registrována na serveru, dojde k jejich zkopírování na server do určitého umístění, které poté není dostupné pro modifikace (stejně jako při zvolení možnosti kopírovat data).

Dále je zadáno umístění na zvoleném hostujícím serveru a vzhledem k účelu aplikace by mělo být nastaveno sdílení se všemi členy GIS serveru geografické sekce PřF UK. Na záložce „Konfigurace“ je ponechána nezaškrtnuta možnost nahrání vstupu uživatelem, protože žádné takové vstupy nástroj nepřijímá. Je zvolen asynchronní režim zpracování; trvá totiž delší dobu, než nástroj vygeneruje výstupy, a rovněž se předpokládá v budoucnosti možnost rozšíření databáze a tím pádem zvýšení množství dat, se kterými bude manipulováno. Také zde lze nastavit úroveň vypisovaného hlášení. Na záložce „Obsah“ je vhodné v konfiguraci GP nástroje doplnit popisový aparát samotného nástroje i jeho jednotlivých parametrů. Pak je před samotnou publikací potřeba nechat budoucí web tool zvalidovat (Analyzovat), a když proběhne kontrola bez problémů, může být nástroj publikován (Ortego, 2018). Pro úpravy skriptu či vlastností nástroje musí být pak vždy aktualizovaná verze publikována na server přepsáním (update) stávajícího webového nástroje.

## 5.2.3 Oprávnění

V souvislosti s publikováním platí pro verzi ArcGIS Serveru 10.4 a vyšší podmínka, že člen organizace musí mít roli administrátora (Administrator), aby mohl publikovat webové nástroje (web tools). Administrátor ale také může pomocí změny nastavení všem členům s rolí vydavatele (Publisher) umožnit publikování geoprocessingových nástrojů. Takto obecné nastavení se ale nedoporučuje kvůli bezpečnosti. Další možností je také využití vlastní role, kterou může administrátor vytvořit a přidělit kterémukoliv členu organizace. Tato vlastní role je odvozena od

některé z existující výchozí role omezením nebo rozšířením některých práv – v případě publikace webového nástroje by tedy mohlo jít např. o roli vycházející z role administrátora, ale s odebranými ostatními právy, které má jedině administrátor, právě kromě možnosti publikace geoprocessingových služeb. Vytvářet vlastní role a nastavovat jejich práva může pouze člen s výchozí rolí administrátora, některá práva týkající se správy ale náleží pouze výchozím administrátorům. Měnit role (kromě role výchozího administrátora) může kdokoliv, kdo má k tomu povolenou pravomoc. Každý uživatel má přiřazen také tzv. typ uživatele, podle kterého je omezeno, kterou může člen zastávat, a to tak, že role uživatele nesmí překročit práva, která se pojí s typem uživatele, který má jeho účet přiřazen (ESRI, 2021ak; ESRI, 2021al; ESRI, 2021al).

### 5.3 Uživatelské rozhraní

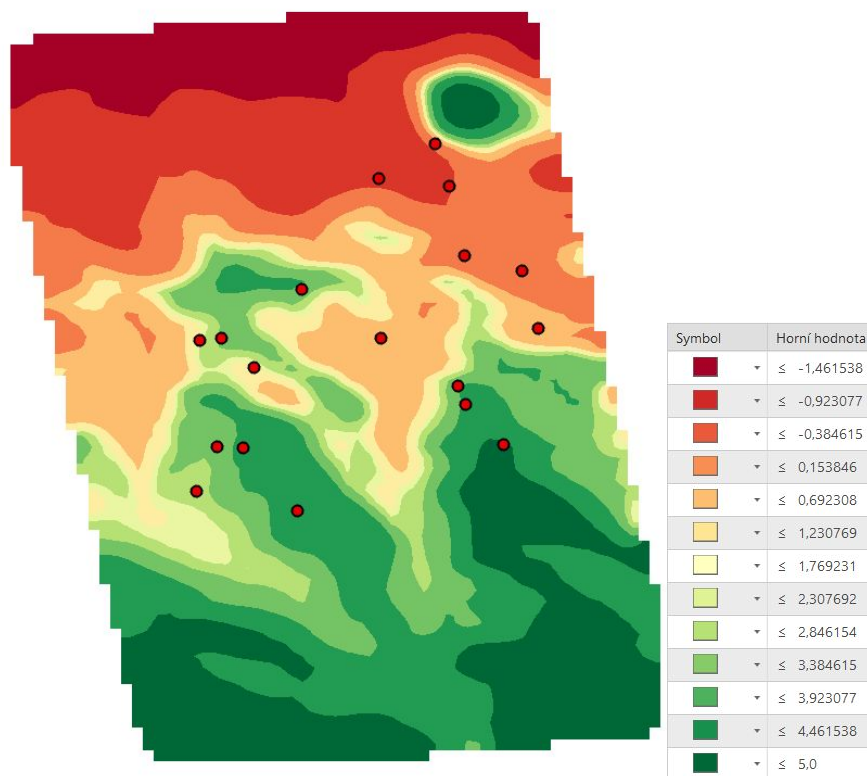
Navrhované uživatelské rozhraní aplikace může být tvořeno v prostředí aplikace ArcGIS Web AppBuilder, které je součástí Portálu. Po úspěšném nastavení prostředí je potřeba zvolit z menu vytvoření nové 2D aplikace. Když je základní aplikace vytvořena, je možné konfigurovat její nastavení. Na záložce „Theme“ je definován vzhled prostředí aplikace, na záložce „Map“ pak podkladovou mapu a její zobrazený rozsah při spuštění aplikace. Záložka „Attribute“ umožňuje další přizpůsobení vzhledu, a zejména také možnost přidat další vrstvy pro zobrazení v aplikaci. Pod záložkou s názvem „Widget“ je umístěno nastavení widgetů. Pro implementaci GP služby publikované na server je ideální využít vzorového widgetu s názvem Geoprocessing. Aby byla nastavena geoprocessingová úloha, je potřeba zadat URL adresu vytvořené služby. Tu je možné nalézt v obsahu Portálu, kde je položka publikované služby zobrazena. Po validaci služby je možné definovat např. popisky a viditelnost vstupních parametrů v UI (u neviditelných je pak použita výchozí hodnota, která je u ostatních pouze nastavena jako první volba), a také symbologii výstupů nástroje. Dále je možné povolit zobrazování pop-up okna včetně jeho obsahu. Výsledné vrstvy lze nechat zobrazit v rámci mapové služby, a též výsledky exportovat, např. do formátu CSV (ESRI 2021ap). Poté musí být nastavení aplikace uloženo. Formulář pro zadávání parametrů bude pak mít nastavenou automatickou kontrolu správného formátu vstupu, korespondující typy parametrů script tool specifikovanými před publikací nástroje.

Pro sdílení aplikace je potřeba ji nejprve ve WAB ze seznamu vytvořených aplikací stáhnout jako komprimovaný soubor, který se po dekompresi umístí na server, upraví jeho konfigurace a aplikace se na server registruje (ESRI 2021aq).

## 6 Výsledky

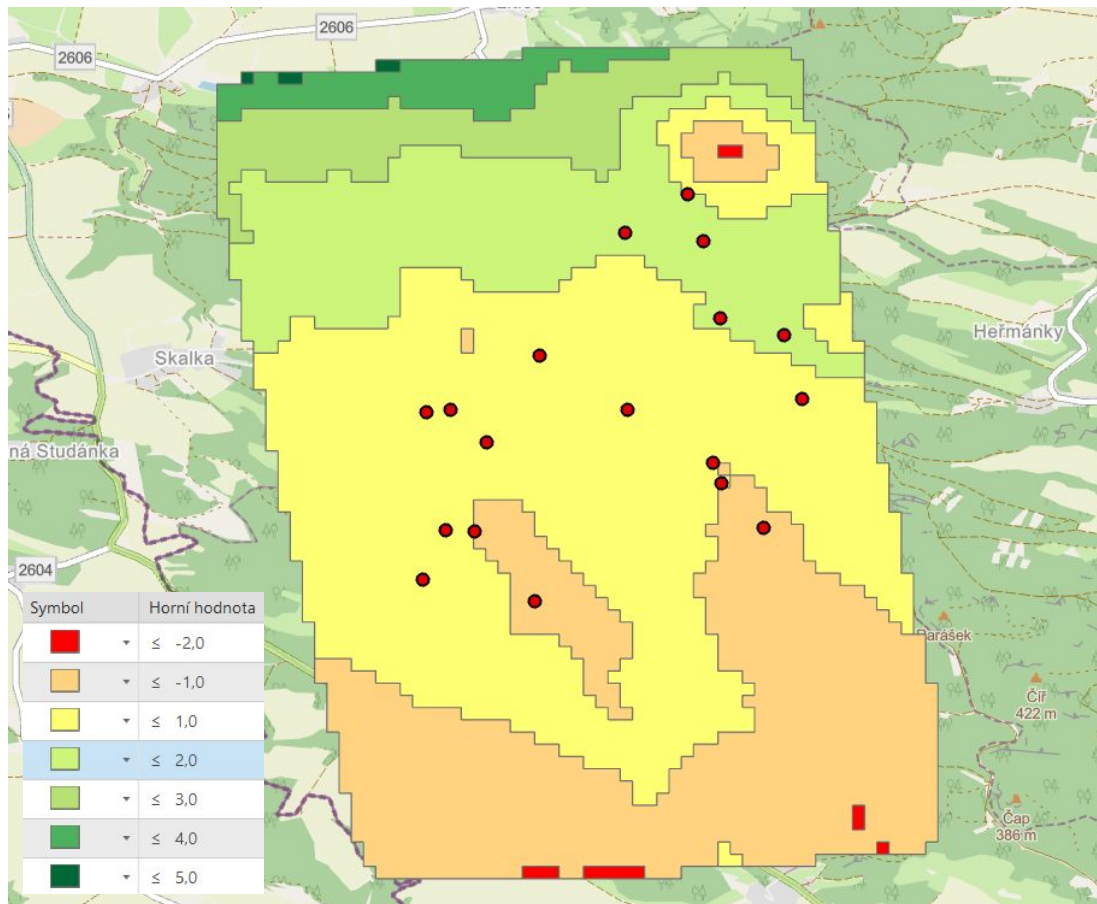
Výsledkem této práce je navržení a vytvoření geoprocessingového nástroje zpracovávajícího zadaná data, aplikujícího analýzy a generujícího výsledky. Z letokruhových chronologií jsou vypočítány mediány Cropperových hodnot, které jsou pak aproximovány do sítě lokalit a interpolovány na zvoleném území. Dále je výsledkem návrh dalšího postupu při tvorbě webové aplikace s přizpůsobeným widgetem, do kterého má být připojena služba, která by vznikla publikací vytvořeného nástroje. Nástroj generuje následující data:

1. vektorovou vrstvu bodů stanovišť s doplněnými údaji z databáze, vysvětlujícími proměnnými, vypočítanými C-hodnotami, jejich predikcemi vycházejícími z interpolační funkce a odchylkami predikovaných hodnot od původních (Obrázek 7)
2. vektorovou vrstvu bodů stanovišť s vysvětlujícími proměnnými, vypočítanými C-hodnotami, predikcemi dle regresní funkce, jejich reziduály, standardizované reziduály a grafové výstupy regresní funkce
3. vektorovou vrstvu pravidelné sítě bodů s vysvětlujícími proměnnými a predikovanými mediány C-hodnot
4. rasterovou vrstvu interpolovaných C-hodnot na základě pravidelné bodové sítě (Obrázek 7 i 8)



Obrázek 7: Příklad rasterového výstupu nástroje se symbolologií nastavenou v ArcGIS Pro (predikce Cropperových hodnot) a bodů lokalit

5. rasterovou vrstvou interpolovaných C-hodnot – reklasifikovanou
6. vektorovou vrstvou obsahující polygony kopírující reklasifikované třídy rasteru interpolovaných C-hodnot (Obrázek 8)



Obrázek 8: Příklad reklasifikovaného rasteru interpolovaných Cropperových hodnot, imitující symbologii (v ArcGIS Pro) navrhovanou v Příloze 2, a bodů lokalit (podkladová mapa: Openstreetmap.org)

Z těchto dat je jako výstupní parametr nástroje definována bodová vrstva stanovišť ad. 1, rasterová vrstva interpolovaných C-hodnot ad. 4 a vektorová polygonová vrstva ad 6.

Vytvořený skript GP nástroje je připojen v Příloze 1.

## 7 Diskuze

Bodovou vrstvu lokalit pro aproximaci C-hodnot by bylo možné vytvořit vždy v průběhu běhu skriptu, ale pro současnou funkcionalitu v rámci této případové studie to není potřeba a také by to bylo nevýhodné z časového hlediska, protože by musela být tato vrstva nejen vždy znovu vygenerována, ale i naplněna všemi potřebnými daty. Na druhou stranu by takovéto řešení přineslo uživateli možnost vlastního nastavení parametru rozestupu bodů, čímž by mohl určit, nakolik budou generalizována data vysvětlující odhadovanou charakteristickou veličinu. Pak by záleželo na uvážení uživatele, aby byl zadán smysluplný údaj. V případě příliš velkých rozestupů bude pak následná interpolace zcela nerelevantní, protože bude zanedbána lokální variabilita vysvětlujících charakteristik, v případě příliš malých vzdáleností bude nejspíše vysoká výpočetní náročnost, a tedy i potřebný čas.

V průběhu vývoje geoprocessingu bylo testováno i pro menší vzdálenosti bodů vstupujících do regresní funkce, i pro větší plochu území (celou Českou republiku), kde ale nastaly problémy s pamětí, z důvodu výpočetní náročnosti, proto byl geoprocessing omezen jen na menší oblast, pro účely případové studie. Ačkoliv odchylky aproximace se budou pravděpodobně s rostoucím množstvím dat ve zdrojové databáze menšit, výpočty pro velké oblasti budou stále časově náročné. Jak již bylo zmíněno, bylo by možné pro celou Českou republiku udělat možnost výběru podle menší územní jednotky, např. polygonu kraje/okresu nebo jiného územního celku, za předpokladu dostatečného množství hodnot vstupujících do regrese. Při současném množství dostupných dat byly totiž výsledky analýz pro celou ČR silně zkreslené. Jednou z možností je také před analýzou přepočítat vstupující vysvětlující proměnné koeficientem vlivu, který podle odhadu na vysvětlovanou veličinu může mít. Obecně se také dá očekávat, že na okrajích oblasti interpolace může docházet k větším odchylkám, z důvodu nevyváženosti množiny bodů vstupujících do interpolačního výpočtu. Podle komentáře doc. Tremly<sup>14</sup> souvisí relevance regresní analýzy výrazně s počtem záznamů vstupujících do regrese a na typu analýzy, kdy např. regresní interpolace podle např. tří vysvětlujících proměnných (nástroj jich nabízí k výběru až pět) by vyžadovala záznamů velké množství. Už při využití dvou proměnných by byl počet 20 záznamů, což je maximální možný počet v případě dostupných testovacích dat, nedostatečný.

Ve skriptu byla použita reklasifikace s pevným krokem, daty byly ovlivněny pouze okrajové třídy. S tím se pojí výhoda možnosti jednoduchého porovnání vektorových výstupů analýzy

---

14 doc. Mgr. Václav Tremly, Ph.D., e-mailové sdělení 10. 6. 2021

s různými parametry mezi sebou. Kdyby se podle vstupního rasteru generoval vždy nový krok pro každý běh nástroje, nevýhodou by bylo to, že nelze porovnávat mezi sebou polygonové výstupy, kvůli odlišné stupnici. Nicméně výhodou dynamického kroku by byla v tom, že v závislosti na různě velkém vzorku vybraných dat se maximální a minimální hodnoty interpolovaného rasteru mohou výrazně lišit mezi jednotlivými běhy, čemuž by statická reklasifikace nepostačovala, ale dynamická by přizpůsobila meze tříd rozsahu datasetu. Krok by také mohl být definován uživatelem jako parametr. Nejzásadnější je ovšem zachování zřejmosti kladných a záporných extrémů odhadovaných C-hodnot. Reklasifikace a symbologie výstupních vrstev by také mohla vycházet přímo z některých statisticky specifikovaných hranic tříd definujících extrémní roky, podle hodnot, jaké uvádějí například Jetschke et al. ve svém článku z roku 2019 (Jetschke, van der Maaten, van der Maaten-Theunissen, 2019) např. přímo podle Croppera (1979), podle odvození od procentuálních hranic dle Neuwirtha et al. (2007) nebo podle jiného arbitrárně zvoleného ohraničení závislého na procentuálním podílu.

Nyní získává nástroj souřadnice z databáze, ale rasterové hodnoty z rasterových Imagery vrstev, protože v databázi nejsou všude tyto hodnoty doplněné. Když by v budoucnu byly doplněny, bylo by možné zdroj přesměrovat a rasterová data ze serveru pak nebudou potřeba. Při doplnění klimatických dat do databáze a v rasterové podobě a adekvátní úpravě skriptu může nástroj zpracovávat i dendroklimatologickou analýzu. Ačkoliv při současném stavu databáze se v ní vyskytují pouze jednodruhové sady vzorků, dá se předpokládat budoucí potřeba rozlišovat záznamy vstupující do analýzy podle toho, z jakého druhu stromu měření pochází, čehož může být docíleno přidáním daného parametru do vlastností nástroje a jeho zařazením do SQL dotazu. Kdyby bylo ale více druhů měřených v rámci jednoho stanoviště, nebylo by možné uvádět v případné výstupní tabulce stanovišť údaj o druhu. V současnosti je také možnost výběru dat omezena na ne/detrendovaný dataset, ale v budoucnu by bylo možné aplikaci rozšířit o možnost nastavitelné úrovně detrendace.

S vývojem většiny prvků skriptu bylo nutné počkat do doby, kdy bude známá struktura databáze (zejména názvy a obsah polí), a kdy k ní bude umožněn přístup pomocí SQL dotazu. S nastavením frontendu pomocí Geoprocessing widgetu je zase možné začít až po úspěšné publikaci GP služby. Při zpracování skriptu nastaly také komplikace. Během testování první verze nástroje před publikací na server nastal problém pádu nástroje vždy ve fázi první funkce podávající požadavek na obrazová data na serveru. Tato chyba byla zapříčiněna ukončením platnosti i certifikátu webového rozhraní Portálu, které se odehrálo ve stejnou chvíli, od které

přestal fungovat skript, a zpočátku byl problém i s připojením k samotnému rozhraní. Poté, co započala platnost nového certifikátu, bylo vše opět zprovozněno, nicméně tato situace načas zablokovala vývoj a pozdržela testování geoprocessingu. Později bylo zjištěno, že podobný problém nastal i v případě webové stránky Server Manageru, kde rovněž vypršela platnost certifikátu zabezpečení. Stránky pak měly problém s načítáním a přístup na ně byl možný jen s využitím anonymního režimu prohlížeče.

V další fázi vývoje při pokusu o publikaci nástroje jako GP služby s odkazováním na registrovaná data se objevila chyba, která zablokovala postup vytváření aplikace, a kvůli které nebyla část popisované metodiky uvedena do praxe (zejména kvůli nemožnosti propojit GP s uživatelským prostředím aplikace). Podle hlášky oznamující poškozený zdroj projektových dat a po vyloučení dalších možných příčin byla jako důvod této chyby identifikována právě dendrochronologická databáze, respektive problém při přístupu k ní. Přestože jsou sdíleny na portál, nejsou rasterové podklady a hostovaná bodová vrstva registrovány do interního registru serveru. Zvolení kterékoliv z možností odkazovat na projektová data nebo je zkopírovat tedy vždy vede pouze ke zkopírování, jak bylo zmíněno v podkapitolách 5.1.2 a 5.2.2. V případě databáze ale nebylo možné publikovat službu s nastavením zkopírování dat. Pro to, aby měl ArcGIS Server přístup k databázi, je potřeba, aby byl SDE soubor připojení k databázi registrován na serveru. Tuto registraci, jak bylo zjištěno, může uskutečnit správce, který má přístup k adresářové struktuře serveru. Lze očekávat, že při úspěšné registraci konfiguračního SDE souboru přestane analýza (validace) nástroje před publikací hlásit chybu poškozeného datového zdroje a následně bude moci být dokončen vývoj navrhované webové aplikace.

## 8 Závěr

Tato bakalářská práce se zabývala návrhem a tvorbou geoprocessingového nástroje sestaveného v jazyce Python. Tento skript zpracovává letokruhové chronologie a počítá z nich mediány Cropperových hodnot, jejichž hodnoty jsou pomocí regresní funkce aproximovány do bodové sítě. Aproximované hodnoty jsou následně interpolovány na území oblasti případové studie (sever CHKO Kokořínsko). Dále byl v rámci této práce popsán návrh implementace vytvořeného nástroje jako webové služby do přednastaveného widgetu, jeho konfigurace, a tvorba webové aplikace obsahující tento widget v prostředí Web AppBuilder. Kvůli technickým komplikacím nebylo možné provést praktickou aplikaci navrhovaného propojení script tool s widgetem a webovou aplikací, nicméně v budoucnosti by měla být po publikaci skriptu aplikace propojena webovou službou podle popisované metodiky. Práce poskytuje podrobný přehled postupů, které je třeba zařadit při vývoji aplikace, jejímž účelem je zpracování specifického geoprocessingu a prezentace jeho výsledků.

Metodika této práce může přispět při technologickém vývoji v úvodu zmíněného projektu Technologické agentury České republiky, který má na starosti KFGG. Cílem projektu je mimo jiné hodnocení a predikce reakce dřevin na klimatické vlivy; tato reakce může být indikována i výkyvy v rovnoměrnosti letokruhových přírůstků, charakterizovanými Cropperovými hodnotami, které vytvořený skript počítá. Také může uvedená metodika nebo její část posloužit jako jeden z podpůrných prvků pro zpracování některého z dílčích záměrů tohoto projektu týkajících se například tvorby webové platformy prezentující výsledky výstupů projektu formou webových GISových vrstev a map.



## 9 Použité zdroje

- ARCDATA 2017. *Práce s webovými službami v ArcGIS Pro*. ARCDATA PRAHA, s.r.o. [online]. Dostupné z [https://www.youtube.com/watch?v=\\_ycDFBpcAI0&t=47s](https://www.youtube.com/watch?v=_ycDFBpcAI0&t=47s).
- ARCDATA PRAHA, ZÚ, ČSÚ (2014): ArcČR 500 - digitální geografická databáze, verze 3.3. [www.arcdata.cz](http://www.arcdata.cz). [cit. 2019-4-3]
- ASPARA 2015. Prvky kmene na příčném řezu. *ASPARA: Poradenství, služby a vzdělávání pro dřevařský a nábytkářský průmysl*. Dotováno v roce 2015. [online]. [cit. 2021-7-26]. Dostupné z: <http://www.elearning.aspara.cz/subdom/elearning/index.php?page=stavba-dreva>
- CLIFF, Andrew David; ORD, John Keith. *Spatial autocorrelation: Monographs in spatial and environmental systems analysis*. 5. London: Pion, 1973. ISBN 0-85086-036-9.
- *Co je to server?* IT Slovník [online]. b.r. [cit. 2021-7-27]. Dostupné z: <https://it-slovník.cz/pojem/server>
- CROPPER, John Philip. *Tree-ring skeleton plotting by computer*. [online]. 1979. Tree-Ring Bulletin (39) 47-60. Dostupné z:
- ČÚZK 2010. Digitální model reliéfu České republiky 4. generace (DMR 4G) ČÚZK [online]. 2010. [cit. 2021-5-24]. Dostupné z: [https://geoportal.cuzk.cz/\(S\(zpbvr3hc0zr2dsvrix2tkfe4\)\)/Default.aspx?mode=TextMeta&side=vyskopis&metadataID=CZ-CUZK-DMR4G-V&head\\_tab=sekce-02-gp&menu=301](https://geoportal.cuzk.cz/(S(zpbvr3hc0zr2dsvrix2tkfe4))/Default.aspx?mode=TextMeta&side=vyskopis&metadataID=CZ-CUZK-DMR4G-V&head_tab=sekce-02-gp&menu=301).
- *Dendroarcheologie*. Dendrochronologie.cz. [online]. Copyright ©2021. [cit. 2021-7-25]. Dostupné z: <http://dendrochronologie.cz/cs/o-dendrochronologii/dendroarcheologie/>
- *Dendroekologie*. Dendrochronologie.cz. [online]. Copyright ©2021. [cit. 2021-7-25]. Dostupné z: <http://dendrochronologie.cz/cs/o-dendrochronologii/dendroekologie/>
- DO PRADO, Kelvin Salton. Understanding time complexity with Python examples. *Towards Data Science* [online]. 4. 3. 2019. [cit. 2021-6-13]. Dostupné z: <https://towardsdatascience.com/understanding-time-complexity-with-python-examples-2bda6e8158a7>
- DUGGAN, Nicholas. *Web Mapping for Dummies – My Personal Experience*. [online]. 2014 Dostupné z URL: <https://www.xyht.com/spatial-itgis/web-mapping-for-dummies-my-personal-experience/>
- ESRI 2021a. *Imagery layers, tile caches, and map image layers*. ArcGIS Online. ESRI [online]. 2021. [cit. 2021-7-17] Dostupné z:

<https://doc.arcgis.com/en/arcgis-online/reference/map-image-elevation-imagery-layers.htm>.

- ESRI 2021aa. *Search neighborhoods*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-20]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/help/analysis/geostatistical-analyst/search-neighborhoods.htm>.
- ESRI 2021ab. Draw widget. ArcGIS Web AppBuilder. *ESRI* [online]. 2021. [cit. 2021-7-4]. Dostupné z: <https://doc.arcgis.com/en/web-appbuilder/create-apps/widget-draw.htm>
- ESRI 2021ac. *Calculate Field (Data Management)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-5-8]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/data-management/calculate-field.htm>.
- ESRI 2021ad. *Calculate Field Python examples*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-5-9]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/data-management/calculate-field-examples.htm>.
- ESRI 2021ae. *RemapRange*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-5-16]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/spatial-analyst/an-overview-of-transformation-classes.htm>.
- ESRI 2021af. *SetParameterSymbology*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-23]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/functions/setparametersymbology.htm>.
- ESRI 2021ag. *Renderer objects*. ArcGIS Developers. *ESRI* [online]. 2021. [cit. 2021-7-23]. Dostupné z: <https://developers.arcgis.com/documentation/common-data-types/renderer-objects.htm>.
- ESRI 2021ah. *Simple Fill Symbol (esriSFS)*. ArcGIS Developers. *ESRI* [online]. 2021. [cit. 2021-7-23]. Dostupné z: [https://developers.arcgis.com/web-map-specification/objects/esriSFS\\_symbol/](https://developers.arcgis.com/web-map-specification/objects/esriSFS_symbol/).
- ESRI 2021ai. *Understanding message types and severity*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-29]. Dostupné z: [https://pro.arcgis.com/en/pro-app/2.6/arcpy/geoprocessing\\_and\\_python/message-types-and-severity.htm](https://pro.arcgis.com/en/pro-app/2.6/arcpy/geoprocessing_and_python/message-types-and-severity.htm).
- ESRI 2021aj. *Setting script tool parameters*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-26]. Dostupné z: [https://pro.arcgis.com/en/pro-app/2.6/arcpy/geoprocessing\\_and\\_python/setting-script-tool-parameters.htm](https://pro.arcgis.com/en/pro-app/2.6/arcpy/geoprocessing_and_python/setting-script-tool-parameters.htm).

- ESRI 2021ak. *Change geoprocessing service and service extension publishing privileges*. ArcGIS Server 10.8. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://enterprise.arcgis.com/en/server/10.8/administer/windows/change-geoprocessing-service-and-service-extension-publishing-privileges.htm>.
- ESRI 2021al. *User types, roles, and privileges*. Portal for ArcGIS 10.8. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://enterprise.arcgis.com/en/portal/10.8/administer/windows/roles.htm>.
- ESRI 2021am. *Change geoprocessing service and service extension publishing privileges*. ArcGIS Server 10.8. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://enterprise.arcgis.com/en/server/10.8/administer/windows/change-geoprocessing-service-and-service-extension-publishing-privileges.htm>.
- ESRI 2021an. *Web maps—ArcGIS Online Help | Documentation*. ArcGIS [online]. [cit. 2021-7-27]. Dostupné z: <https://doc.arcgis.com/en/arcgis-online/reference/what-is-web-map.htm>
- ESRI 2021ao. *What is ArcGIS Web AppBuilder?* ArcGIS [online]. [cit. 2021-7-28]. Dostupné z: <https://doc.arcgis.com/en/web-appbuilder/create-apps/what-is-web-appbuilder.htm>
- ESRI 2021ap. *Geoprocessing widget*. ArcGIS [online]. [cit. 2021-7-28]. Dostupné z: <https://developers.arcgis.com/web-appbuilder/guide/widget-geoprocessing.htm>
- ESRI 2021aq. *Deploy your app*. ArcGIS [online]. [cit. 2021-7-28]. Dostupné z: <https://developers.arcgis.com/web-appbuilder/guide/xt-deploy-app.htm>
- ESRI 2021b. *User types, roles, and privileges*. ArcGIS 10.9 (Windows). *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://enterprise.arcgis.com/en/portal/latest/administer/windows/roles.htm>.
- ESRI 2021c. *Introduction to sharing web layers*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-18]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/help/sharing/overview/introduction-to-sharing-web-layers.htm>.
- ESRI 2021d. *Share a web feature layer*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-28]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/help/sharing/overview/web-feature-layer.htm>.
- ESRI 2021e. *Accessing data sources in Python*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-13]. Dostupné z:

[https://pro.arcgis.com/en/pro-app/2.6/arcpy/geoprocessing\\_and\\_python/data-sources-in-python.htm](https://pro.arcgis.com/en/pro-app/2.6/arcpy/geoprocessing_and_python/data-sources-in-python.htm).

- ESRI 2021f. *What is the Data Access module*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/data-access/what-is-the-data-access-module-.htm>.
- ESRI 2021g. *What is the Spatial Analyst module*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/spatial-analyst/what-is-the-spatial-analyst-module.htm>.
- ESRI 2021h. *What is the Geostatistical Analyst extension in ArcGIS Pro?* ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/help/analysis/geostatistical-analyst/what-is-the-geostatistical-analyst-extension-in-arcgis-pro.htm>.
- ESRI 2021i. *An overview of the Data Management toolbox*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/data-management/an-overview-of-the-data-management-toolbox.htm>.
- ESRI 2021j. *An overview of the Conversion toolbox*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/conversion/an-overview-of-the-conversion-toolbox.htm>.
- ESRI 2021k. *An overview of the Spatial Statistics toolbox*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/spatial-statistics/an-overview-of-the-spatial-statistics-toolbox.htm>.
- ESRI 2021l. *Scratch Folder (Environment setting)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-18]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/environment-settings/scratch-folder.htm>.
- ESRI 2021m. *Scratch GDB (Environment setting)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-18]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/environment-settings/scratch-gdb.htm>.
- ESRI 2021n. *Write geoprocessing output to memory*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-19]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/help/analysis/geoprocessing/basics/the-in-memory-workspace.htm>.

- ESRI 2021o. *Quick tour of authoring and sharing geoprocessing services*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-12]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/help/analysis/geoprocessing/share-analysis/quick-tour-of-authoring-and-sharing-geoprocessing-services.htm>.
- ESRI 2021p. *Create Database Connection (Data Management)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-29]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/data-management/create-database-connection.htm>.
- ESRI 2021q. *SearchCursor*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-19]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/functions/searchcursor.htm>.
- ESRI 2021r. *Create Feature Class (Data Management)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-21]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/data-management/create-feature-class.htm>.
- ESRI 2021s. *SpatialReference*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-21]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/classes/spatialreference.htm>.
- ESRI 2021t. *Add Field (Data Management)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-5-8]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/data-management/add-field.htm>.
- ESRI 2021u. *Write geometries*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-15]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/get-started/writing-geometries.htm>.
- ESRI 2021v. *InsertCursor*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-22]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/functions/insertcursor.htm>.
- ESRI 2021w. *UpdateCursor*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-21]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/data-access/updatecursor-class.htm>.
- ESRI 2021x. *Accessing data using cursors*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-7-15]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/arcpy/get-started/data-access-using-cursors.htm>.
- ESRI 2021y. *Generalized Linear Regression (GLR) (Spatial Statistics)*. ArcGIS Pro 2.6. *ESRI* [online]. 2021. [cit. 2021-6-3]. Dostupné z: <https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-statistics/generalized-linear-regression.htm>.

- ESRI 2021z. *Local Polynomial Interpolation (Geostatistical Analyst)*. ArcGIS Pro 2.6. ESRI [online]. 2021. [cit. 2021-5-26]. Dostupné z: <https://pro.arcgis.com/en/pro-app/2.6/tool-reference/geostatistical-analyst/local-polynomial-interpolation.htm>.
- FETCHER, Josh. *15 Common Software as a Service (SaaS) Examples to Inspire You*. 2021. [online]. Joshfetcher.com. [cit. 2021-7-28]. Dostupné z: <https://joshfechter.com/software-service-examples/>
- HAO, HE: *What Is Service-Oriented Architecture*. 2003. [online]. xml.com. O'Reilly Media. [cit. 2021-7-27]
- HOLEC, Miroslav. *Historie REST API*. 2021. [online]. Školení a konzultace REST API a .NET 5. Miroslavholec.cz. [cit. 2021-7-28]. Dostupné z: <https://www.miroslavholec.cz/blog/historie-rest-api>
- HOLUBEC, Vladimír. *Práce s webovými službami v ArcGIS Pro*. 2017. YouTube.com. [online]. Publikováno 18. 10. 2017 [cit. 2021-06-13]. Dostupné z: [https://www.youtube.com/watch?v=\\_ycDFBpcAI0](https://www.youtube.com/watch?v=_ycDFBpcAI0)  
<https://repository.arizona.edu/bitstream/handle/10150/260438/trb-39-047-059.pdf?sequence=1&isAllowed=y>
- How To Create Range Sliders. W3Schools Online Web Tutorials [online]. Dostupné z: [https://www.w3schools.com/howto/howto\\_js\\_rangeslider.asp](https://www.w3schools.com/howto/howto_js_rangeslider.asp)
- IBM Cloud Education. *What is SOA (Service-Oriented Architecture)?* 2019. IBM. [online]. [cit. 2021-7-27]. Dostupné z: <https://www.ibm.com/cz-en/cloud/learn/soa>
- JETSCHKE, Gottfried, VAN DER MAATEN, Ernst; VAN DER MAATEN-THEUNISSEN, Marieke . Towards the extremes: a critical analysis of pointer year detection methods. *Dendrochronologia* [online]. 2019, 2019(53), 55–62 [cit. 2021-3-21]. ISSN 1125-7865. Dostupné z: doi:<https://doi.org/10.1016/j.dendro.2018.11.004>
- JEŽEK, Josef. *Geostatistika a prostorová interpolace*. v Praze: Univerzita Karlova, nakladatelství Karolinum, 2015. ISBN 978-80-246-3076-2.
- KŘIKAVOVÁ, Linda. *Interpolace bodových dat v GIS*. Praha, 2009. Bakalářská práce. České vysoké učení technické v Praze, Fakulta stavební, Katedra mapování a kartografie. Vedoucí práce Cajthaml, Jiří.
- LEŠTINA, Petr. *Co je Servisně Orientovaná Architektura?* 2007. BPM a IBM [online]. ISSN 1802-5675. Dostupné z: <http://bpm-ibm.blogspot.com/2007/11/co-je-servisn-orientovan-architektura.html>

- LYSÁK, Jakub. Čtení geometrie v Pythonu [online přednáška]. Praha: Přírodovědecká fakulta UK, 1. dubna 2021. Záznam dostupný z: <https://drive.google.com/file/d/1rNz98aJJuJL1oBB8oLqGz70Dq7RVXxIk/view>
- MAJER, Hugo-Miroslav. *Návrh databáze letokruhových chronologií*. Praha, 2021. Bakalářská práce. Univerzita Karlova, Přírodovědecká fakulta, Katedra aplikované geoinformatiky a kartografie. Vedoucí práce Brůha, Lukáš.
- ManagementMania. *Aplikační server (Application server)*. 2020. ManagementMania.com. [online]. [cit. 2021-7-27]. Dostupné z: <https://managementmania.com/cs/aplikacni-server-aps>
- MARGARIS, Nikos. *Cloud Computing v českém prostředí*. 2011. [online]. Diplomová práce. Vysoká škola ekonomická v Praze. [cit. 2021-7-27]. Dostupné z: <https://vskp.vse.cz/28381>
- MASON, Ross. *How REST replaced SOAP on the Web: What it means to you*. 2011. [online]. InfoQ. [cit. 2021-7-28]. Dostupné z: <https://www.infoq.com/articles/rest-soap/>
- MCINTOSH, Bryan. Use ArcPy to Connect to SDE and Versions. *Spatial Times* [online]. Canada, 16 Nov 2018. [cit. 2021-6-29]. Dostupné z: <https://www.spatialtimes.com/2018/10/use-arcpy-to-connect-to-sde-and-versions/>
- *Metodika venkovního sběru dat* [online]. b.r. Ústav pro hospodářskou úpravu lesů Brandýs nad Labem [cit. 2021-7-11]. Verze 6.0. Dostupné v archivu: [https://web.archive.org/web/20110409141510/http://www.uhul.cz/il/metodika/metodika6/kap\\_3\\_6\\_0.pdf](https://web.archive.org/web/20110409141510/http://www.uhul.cz/il/metodika/metodika6/kap_3_6_0.pdf) pořízeném dne 2011-04-09.
- MÍK, Milan. *Geografická charakteristika extrémních propadů růstu borovice lesní ve střední Evropě*. Praha, 2020. Diplomová práce. Univerzita Karlova, Přírodovědecká fakulta, Katedra fyzické geografie a geoekologie. Vedoucí práce Tremml, Václav.
- NÉTEK, Rostislav. *Webová kartografie: specifika tvorby interaktivních map na webu*. Olomouc: Univerzita Palackého v Olomouci, 2020. ISBN 9788024458274.
- NEUMANN, Andreas. *Web Mapping and Web Cartography*. [online]. 2017. In: Shekhar S., Xiong H., Zhou X. (eds) *Encyclopedia of GIS*. Springer, Cham. [cit. 2021-7-27] DOI: 10.1007/978-3-319-17885-1\_1485
- NEUWIRTH, Burkhard; SCHWEINGRUBER, Fritz; WINIGER, Matthias. Spatial patterns of central European pointer years from 1901 to 1971. 2007. *Dendrochronologia*. (24) 79-89. <https://doi.org/10.1016/j.dendro.2006.05.004>. Dostupné z: [https://www.researchgate.net/publication/229176713\\_Spatial\\_patterns\\_of\\_central\\_European\\_pointer\\_years\\_from\\_1901\\_to\\_1971](https://www.researchgate.net/publication/229176713_Spatial_patterns_of_central_European_pointer_years_from_1901_to_1971)

- ORTEGO, Andrew. *Creating Geoprocessing Services with Python Script Tools*. [video]. Esri Events. Publikováno 30. 4. 2018. [cit. 2021-7-12]. Dostupné z: <https://www.esri.com/videos/watch?videoid=CFp-R03KkfU>
- PICHLÍK, Roman. a REST. *dagblog* [online]. 16. 10. 2007 [cit. 2021-7-16]. Dostupné z: <https://dagblog.cz/a-rest-c5156313d79e>
- Python 2021. *os.path — Common pathname manipulations*. Python 3.9.6 Documentation. 2021. [cit. 2021-7-19]. Dostupné z: <https://docs.python.org/3/library/os.path.html>.
- PythonWiki 2020. The Python Wiki. TimeComplexity. [online] Aktualizace 18. 8. 2020. [cit. 2021-6-7]. Dostupné z: <https://wiki.python.org/moin/TimeComplexity>.
- SACK, Carl M. *Web Mapping*. [online]. 2017. The Geographic Information Science & Technology Body of Knowledge (4th Quarter 2017 Edition), John P. Wilson (ed.). [cit. 2021-7-27] DOI: 10.22224/gistbok/2017.4.11
- SEQUENS, Josef. *Dendrometrie*. 2007. Stránky kateder a útvarů ČZU. [cit. 2021-7-25]. Dostupné z: [https://katedry.czu.cz/storage/3844\\_Souhrn\\_Dendrometrie.pdf](https://katedry.czu.cz/storage/3844_Souhrn_Dendrometrie.pdf)
- SCHWEINGRUBER, Fritz H.; ECKSTEIN, Dieter; SERRE-BACHET, Françoise; BRÄKER, Otto U. *Identification, presentation and interpretation of event years and pointer years in dendrochronology*. 1990. Dendrochronologia (8). Dostupné z: <https://www.naturfagsenteret.no/c1761334/binfil/download2.php?tid=1856970>
- ŠTRÁFELDA, Jan. *Jan Štráfelda: průvodce online projektem: Know-how - Encyklopedie* [online]. b.r. [cit. 2021-7-27]. Dostupné z: [www.strafelda.cz](http://www.strafelda.cz)
- TA ČR 2021. *Databáze letokruhových chronologií jako nástroj pro evidenci a predikci reakce hlavních lesních dřevin na klimatickou změnu (SS03010134): zadávací dokumentace*. Praha : Technologická agentura ČR. Program Prostředí pro život. Řešitel: Václav Tremel. 2021-2024.
- UCHIHA, Madara. *What are the differences between server-side and client-side programming?* 2018. [online]. [cit. 2021-7-27]. Dostupné z: [softwareengineering.stackexchange.com](https://softwareengineering.stackexchange.com)
- USNA 2019. *Approximate metric equivalents for degrees, minutes, and seconds*. United States Naval Academy [online]. 8. 5. 2019. [cit. 2021-6-17]. Dostupné z: [https://www.usna.edu/Users/oceano/pguth/md\\_help/html/approx\\_equivalents.htm](https://www.usna.edu/Users/oceano/pguth/md_help/html/approx_equivalents.htm).
- VA VYŠKOV. *Tvorba webu – HTML, CSS, JS: Webové stránky stručně a prakticky*. [online]. b.r. [cit. 2021-7-27]. Dostupné z: <https://web.vavyskov.cz/>
- VAN REES, Eric. *Populating a Feature Class in ArcGIS Pro using an Insert Cursor in Arcpy*. *GeoSpatial Training Services* [online]. 23. 3. 2021. [cit. 2021-6-22]. Dostupné z:



<https://geospatialtraining.com/populating-a-feature-class-in-arcgis-pro-using-an-insert-cursor-in-arcpy/>

- VRTICH, Matej. Sdílení vektorových dat na ArcGIS Online a ArcGIS Enterprise. *ArcREVUE: Časopis pro uživatele softwaru Esri a ENVI* [online]. Praha: ARCDATA PRAHA, 2018, 2018(4), 34 [cit. 2021-7-14]. ISSN 1211-2135. Dostupné z: <https://www.arcdata.cz/zpravy-a-akce/publikace/arcrevue/archiv-arcrevue/arcrevue-4-2018>
- WENDLAND, Wayne Marcel. *An objective method to identify missing and false rings.* [online]. 1975. *Tree-Ring Bulletin.* (35) 41-47. Dostupné z: <https://repository.arizona.edu/bitstream/handle/10150/260132/trb-35-041-047.pdf?sequence=1&isAllowed=y>
- ZVÁRA, Karel. *Základy statistiky v prostředí R.* IV. 2013. ISBN 9788024622453.

## 10 Seznam příloh

Příloha 1: Skript GP nástroje provádějící výpočet Cropperových hodnot a následnou regresní analýzu a interpolaci, určený pro připojení do widgetu webové aplikace jako GP služba

Příloha 2: Možnost definování symbologie výstupní polygonové vrstvy s využitím JSON rendereru

# Příloha 1: Skript GP nástroje provádějící výpočet Cropperových hodnot a následnou regresní analýzu a interpolaci, určený pro připojení do widgetu webové aplikace jako GP služba

```
##### DENDROCHRONOLOGY - TREE RING ANALYSIS #####

### IMPORT MODULES #####
import arcpy
import os
import tempfile
import shutil
from math import sqrt, floor, ceil
from statistics import median
import collections
import sys
import time
#####

# timer
all_start = time.time()

### ENVIRONMENT #####
# set environment
arcpy.env.overwriteOutput = True
arcpy.env.qualifiedFieldNames = False
#####

### FUNCTIONS #####
def C_values_medians_function(dictionary):
    """Calculates Cropper values & medians for one site.

    Input: dictionary of years and corresponding annual ring growths (dict).
    Output: ordered dictionary of years and corresponding medians of Cropper values (dict).
    """
    w = 13
    k = floor(w/2)
    C_vals = {}
    catalog = {}

    for key,value in dictionary.items(): # item == sample
        C = []
        cy = [] # current year index (year that we base the window in)

        for l in range(len(dictionary[key]['years'])): # create list of usable 'cy's
            cy.append(l)

        for h in range(k):
            cy.remove(h)
        for j in range(k):
            delete = len(value['values'])-1-j
            cy.remove(delete)

        # loops through usable 'cy's
        for i in range(cy[0],cy[-1]+1): # i is current mid-window year
            mid_year = value['values'][i] # value of current mid-window year

            sum_val = 0 # sum of all years' values in window
            for ii in range(-k,k+1):
                sum_val += value['values'][i+ii]
            window_mean = sum_val/(2*k+1) # mean of all years' values in window

            sum_q_dev = 0 # sum of deviations
            for iii in range(-k,k+1):
                sum_q_dev += ((value['values'][i+iii]) - window_mean)**2
            stdev_cy = sqrt(sum_q_dev/(2*k)) # two-direction stdev - both sides (w-1)

            c = round((mid_year - window_mean)/stdev_cy,2) # 2 decimal places
            C.append({value['years'][i]: c})

            year_num = value['years'][i]
            if year_num not in catalog.keys():
                catalog[year_num] = [c]
            else:
                catalog[year_num].append(c)

        C_vals[key] = C

    new_cat = {}
    sorted_catalog = collections.OrderedDict(sorted(catalog.items()))
    for item in sorted_catalog.items():
        if len(item[1]) >= 5: # new_cat does not include years represented by 4 or less samples
            new_cat[item[0]] = round(median(item[1]),2)
    snew_cat = collections.OrderedDict(sorted(new_cat.items()))
    return snew_cat

def newfield(input_table,f_name,f_type,f_scale):
    """Adds new field with given name, type and scale to the input table.
```

```

Input: input table (str), field name (str), field type (str), field scale (number of decimal places) (int).
"""
arcpy.management.AddField(
    in_table=os.path.join(input_table),
    field_name=f_name,
    field_type=f_type,
    field_scale=f_scale)

def range_list(min,max):
    """Creates a list of lists specifying class ranges for reclassification.
    Outer boundaries are according to interpolated raster's min or max value (whichever is absolutely larger).

    Input: minimal raster value (float), maximal raster value (float).
    Output: remap list of lists with class boundaries and values (list), step (class interval) (float).
    """
    if abs(min) > abs(max):
        outer = abs(min)
    else:
        outer = abs(max)

    outer = ceil(outer*100)/100 # round UP to 2 decimal places
    step = 0.15 # fixed step
    border = step*5

    rangelist = []
    # classes from -6 to 6; 0 == 0, -6 & 6 is used for greater than 'border' or '-border', respectively
    if outer > border:
        classList = [-outer,-border,-6] # 1st class if needed
        rangeList.append(classList)

    a=-border
    c=-5

    for _ in range(11):
        b=a+step
        if c == 0:
            classList=[-0.0001,0.0001,c] # zero class
        elif c == -1:
            classList=[round(a,2),-0.0001,c] # next-to-zero class (due to zero's boundaries)
        elif c == 1:
            a=0
            b=a+step
            classList=[0.0001,round(b,2),c] # next-to-zero class (due to zero's boundaries)
        else:
            classList=[round(a,2),round(b,2),c]

        rangeList.append(classList)
        a+=step
        c+=1

    if outer > border:
        classList = [border,outer,6] # last class if needed
        rangeList.append(classList)

    return rangeList,step
#####

### USER SELECTION PARAMETERS from JS API frontend #####
det = arcpy.GetParameterAsText(0) # USER INPUT (bool) - selection of dataset
if det == 'True' or det == 'true':
    data = 'detrend_data'
    dataset_name = "detrendovaná data"
elif det == 'False' or det == 'false':
    data = 'raw_data'
    dataset_name = "nedetrendovaná data"
min_length = arcpy.GetParameterAsText(1) # USER INPUT (int) - minimal chronology length
tree_height_min = arcpy.GetParameterAsText(2) # USER INPUT (float) - tree height range (m)
tree_height_max = arcpy.GetParameterAsText(3) # USER INPUT
cr_height_min = arcpy.GetParameterAsText(4) # USER INPUT (float) - crown height range (m)
cr_height_max = arcpy.GetParameterAsText(5) # USER INPUT
dbh_min = arcpy.GetParameterAsText(6) # USER INPUT (float) - diameter at breast height range
(cm) (average...)
dbh_max = arcpy.GetParameterAsText(7) # USER INPUT

# export parameter
delim = arcpy.GetParameterAsText(8) # USER INPUT (str) - delimiter for the export ";" or
","

log_selection = f"\n----- Zvolené parametry výběru: ----- \n-> dataset: {dataset_name}\n-> minimální délka
chronologie: {min_length}\n-> rozsah výšek stromů: {tree_height_min}-{tree_height_max}\n-> rozsah výšek korun:
{cr_height_min}-{cr_height_max}\n-> rozsah průměrů ve výšce hrudi: {dbh_min}-{dbh_max}\n"
arcpy.AddMessage(log_selection)
arcpy.SetParameterAsText(12, log_selection)
#####

### USER ANALYSIS PARAMETERS from JS API frontend #####
explanatory = arcpy.GetParameterAsText(9) # USER INPUT (list) - explanatory variables elevation/
latitude/longitude/slope/aspect... (=> list of strings)

```

```

# if type(explanatory) == str: # for testing in python terminal only
#     explanatory = explanatory.strip('[]').replace(" ", "").split(',')
polynomial = int(arcpy.GetParameterAsText(10)) # USER INPUT (int) - order of interpolation polynomial
0/1/2/3 note: although the interpolation allows it to be 'float', only previously mentioned options will
be available
selected_year = arcpy.GetParameterAsText(11) # USER INPUT (int) - year to do the regression and
interpolation in

log_analysis = f"\n----- Zvolené parametry analýzy: ----- \n-> zvolený rok: {selected_year}\n->
vysvětlující proměnné (regrese): {str(explanatory).strip('[]')}\n-> stupeň polynomu (interpolace): {polynomial}.\n"
arcpy.AddMessage(log_analysis)
arcpy.SetParameterAsText(13, log_analysis)
#####

### GET INFO ABOUT DATASET #####
connection_file = os.path.join("
", "Users", "Anička", "Škola", "3.LS", "Bakalářka", "DendroApp", "DendroConnect", "DendroConnect.sde")
select_all = f"""
SELECT DISTINCT site_code,site_name,species,lat_site,lon_site,elev_site,slope_site,asp_site,rooting_depth,soil,
tree_id,dbh,height,crown_height,crown_projection,social_status,sapwood,elev_tree,slope_tree,asp_tree,
sample_id,year,value FROM
(
SELECT
v.site_code,v.site_name,v.species,v.lat_site,v.lon_site,v.elev_site,v.slope_site,v.asp_site,v.rooting_depth,v.soil,
v.tree_id,
v.dbh,v.height,v.crown_height,v.crown_projection,v.social_status,v.sapwood,v.lat_tree,v.lon_tree,fgt.elevation AS
elev_tree,
fgt.slope AS slope_tree,fgt.aspect AS asp_tree,v.sample_id,v.year,v.value FROM fg_tree AS fgt RIGHT JOIN
(
SELECT
x.site_code,x.site_name,x.species,x.lat_site,x.lon_site,x.elev_site,x.slope_site,x.asp_site,x.rooting_depth,x.soil,
x.tree_id,
dbh,height,crown_height,crown_projection,social_status,sapwood,ST_Y(t.geom) AS lat_tree,ST_X(t.geom) AS
lon_tree,x.sample_id,x.year,x.value FROM tree AS t INNER JOIN
(
SELECT y.site_code,y.site_name,y.species,y.lat_site,y.lon_site,elevation AS elev_site,slope AS slope_site,aspect AS
asp_site,rooting_depth,soil,y.tree_id,y.sample_id,y.year,y.value
FROM fg_site AS fgs RIGHT JOIN
(
SELECT s.site_code,site_name,species,ST_Y(s.geom) AS lat_site,ST_X(s.geom) AS lon_site,tree_id,sample_id,year,value
FROM site AS s INNER JOIN
(
SELECT * FROM {data} AS d WHERE (site_code,tree_id,sample_id) IN
(SELECT DISTINCT site_code,tree_id,sample_id FROM {data} GROUP BY site_code,sample_id,tree_id HAVING COUNT(*)>=0)
) AS z
ON z.site_code=s.site_code
) AS y
ON y.site_code=fgs.site_code
) AS x
ON x.site_code=t.site_code AND x.tree_id=t.tree_id
) AS v
ON fgt.site_code=v.site_code AND fgt.tree_id=v.tree_id
) AS vse
"""
arcpy.management.MakeQueryLayer(connection_file,"database_all_tableview",query=select_all) # makes TableView
arcpy.management.CopyRows("database_all_tableview", os.path.join(arcpy.env.scratchFolder,"db_all.txt")) # adds
OBJECTID
arcpy.management.CopyRows(os.path.join(arcpy.env.scratchFolder,"db_all.txt"), "database_all") # back to TableView

cursor = arcpy.da.SearchCursor("database_all", ["site_code","sample_id"]) # site_code == row[0], sample_id ==
row[1]
all_sitecodes = []
all_samples = []
for row in cursor:
    if row[0] not in all_sitecodes:
        all_sitecodes.append(row[0])
    if row[1] not in all_samples:
        all_samples.append(row[1])
del cursor
#####

### DENDROCHRONOLOGICAL DATA SELECTION #####
select = f"""
SELECT DISTINCT site_code,site_name,species,lat_site,lon_site,elev_site,slope_site,asp_site,rooting_depth,soil,
tree_id,dbh,height,crown_height,crown_projection,social_status,sapwood,elev_tree,slope_tree,asp_tree,
sample_id,year,value FROM
(
SELECT
v.site_code,v.site_name,v.species,v.lat_site,v.lon_site,v.elev_site,v.slope_site,v.asp_site,v.rooting_depth,v.soil,
v.tree_id,
v.dbh,v.height,v.crown_height,v.crown_projection,v.social_status,v.sapwood,v.lat_tree,v.lon_tree,fgt.elevation AS
elev_tree,
fgt.slope AS slope_tree,fgt.aspect AS asp_tree,v.sample_id,v.year,v.value FROM fg_tree AS fgt RIGHT JOIN
(
SELECT
x.site_code,x.site_name,x.species,x.lat_site,x.lon_site,x.elev_site,x.slope_site,x.asp_site,x.rooting_depth,x.soil,
x.tree_id,
dbh,height,crown_height,crown_projection,social_status,sapwood,ST_Y(t.geom) AS lat_tree,ST_X(t.geom) AS

```

```

lon_tree,x.sample_id,x.year,x.value FROM tree AS t INNER JOIN
(
SELECT y.site_code,y.site_name,y.species,y.lat_site,y.lon_site,elevation AS elev_site,slope AS slope_site,aspect AS
asp_site,rooting_depth,soil,y.tree_id,y.sample_id,y.year,y.value
FROM fg_site AS fgs RIGHT JOIN
(
SELECT s.site_code,site_name,species,ST_Y(s.geom) AS lat_site,ST_X(s.geom) AS lon_site,tree_id,sample_id,year,value
FROM site AS s INNER JOIN
(
SELECT * FROM {data} AS d WHERE (site_code,tree_id,sample_id) IN
(SELECT DISTINCT site_code,tree_id,sample_id FROM {data} GROUP BY site_code,sample_id,tree_id HAVING
COUNT(*)>={min_length})
) AS z
ON z.site_code=s.site_code
) AS y
ON y.site_code=fgs.site_code
) AS x
ON x.site_code=t.site_code AND x.tree_id=t.tree_id
) AS v
ON fgt.site_code=v.site_code AND fgt.tree_id=v.tree_id
) AS vse

WHERE
(dbh>={dbh_min} AND dbh<={dbh_max})

AND (height>={tree_height_min} AND height<={tree_height_max})
AND (crown_height>={cr_height_min} AND crown_height<={cr_height_max})
""""
arcpy.management.MakeQueryLayer(connection_file,"database_selection_tableview",query=select) # makes TableView
arcpy.management.CopyRows("database_selection_tableview", os.path.join(arcpy.env.scratchFolder,"db_selection.txt"))
# adds OBJECTID
arcpy.management.CopyRows(os.path.join(arcpy.env.scratchFolder,"db_selection.txt"), "database_selection") # back
to TableView
arcpy.AddMessage('Data úspěšně načtena z databáze.')
#####

### GET THE DATA #####
# get sitecodes
cursor = arcpy.da.SearchCursor("database_selection", ["site_code","site_name","species","lat_site","lon_site"]) #
site_code == row[0], site_name == row[1], species == [2], lat_site == [3], lon_site == [4]
sitecodes = []
site_packets = []
for row in cursor:
    if row[0] not in sitecodes:
        sitecodes.append(row[0])
        site_packet = (row[0],(row[4],row[3]),row[1],row[2],row[3],row[4]) # site_packet = (site_code,
(longitude, latitude),site_name,species,latitude,longitude)
        site_packets.append(site_packet)
del cursor

n_sites = len(sitecodes)
log_sites_in = f"Počet zahrnutých lokalit z databáze: {n_sites} z {len(all_sitecodes)}\n(Nemusi být ale zahrnuty
všechny jejich vzorky.)"
arcpy.AddMessage(log_sites_in)
arcpy.SetParameterAsText(14,log_sites_in)

missing_sites = [a for a in all_sitecodes if a not in set(sitecodes)]
if len(missing_sites) != 0:
    log_miss = f"Stanoviště zcela eliminovaná filtry: {str(missing_sites).strip(',')}"
    arcpy.AddMessage(log_miss)
    arcpy.SetParameterAsText(15,log_miss)

# separate sites in dictionary; keys == site codes
cursor = arcpy.da.SearchCursor("database_selection", ["site_code", "sample_id", "year", "value"]) # site_code ==
row[0], sample_id == row[1], year == row[2], value == row[3]
sites_dict = {}
for row in cursor: # each row of the table
    for sitecode in sitecodes: # tries all site codes
        if row[0] == sitecode: # if current site code fits a site code in list 'sitecodes'
            if sitecode in sites_dict.keys(): # if key-value exists
                if row[1] in sites_dict[sitecode].keys(): # if current sample_id is a key in the dictionary
                    sites_dict[sitecode][row[1]]['years'].append(int(row[2])) # append year to list 'years'
                    sites_dict[sitecode][row[1]]['values'].append(row[3]) # append value to list 'values'
                else:
                    sites_dict[sitecode][row[1]] = {'years':[int(row[2])], 'values': [row[3]]}
            else: # create key-value
                record = {row[1]: {'years':[int(row[2])], 'values':[row[3]]}} # {sample_id: {'years':[year],
'values':[value]}}
                sites_dict[sitecode] = record # add under sitecode key
del cursor
#####

### GET CALCULATED SITE VALUES INTO DICTIONARY WITH SITECODES #####
# count of sample_id keys of each location
sample_counts = {}
sumnsam = 0
lengths_chrono = []
for sitec in sitecodes:

```

```

nsam = len(sites_dict[sitec].keys())
for g in (sites_dict[sitec].values()):
    lengths_chrono.append(len(g['years']))
sample_counts[sitec] = nsam
sumnsam += nsam
log_samples_n = f"\nPočet zahrnutých vzorků: {sumnsam} z {len(all_samples)}"
arcpy.AddMessage(log_samples_n)
arcpy.SetParameterAsText(16, log_samples_n)
log_chrono = f"Rozsah délek chronologií vzorků: {min(lengths_chrono)}-{max(lengths_chrono)}"
arcpy.AddMessage(log_chrono)
arcpy.SetParameterAsText(17, log_chrono)

# calculate Cropper values
sites_c_dict = {}
for s_key,s_value in sites_dict.items():
    sites_c_dict[s_key] = C_values_medians_function(s_value)
#####

### GET AVAILABLE YEARS RANGE #####
yearmin = []
yearmax = []
for site,chronology in sites_c_dict.items():
    if len(list(chronology.items())) != 0:
        yearmin.append(list(chronology.items())[0][0]) # first year of Cropper value medians chronology
        (because it is sorted) // (dict.items would return an iterable dict view object rather than a list. We need to wrap
the call onto a list in order to make the indexing possible (via StackOverflow))
        yearmax.append(list(chronology.items())[-1][0]) # last year of Cropper value medians chronology
    else:
        print(f"Stanoviště {site} neobsahuje žádné mediány Cropperových hodnot.") # print in terminal only
yearMIN = max(yearmin) # latest first year -> results in only taking into account years that are present in all
sites' chronologies (of sites that went through the filter and was possible to calculate C. values
yearMAX = min(yearmax) # earliest last year -> dtto

log_all_cropp = f"Rozpětí let, ve kterých byly vypočítány mediány Cropperových hodnot pro všechna neodfiltrovaná
stanoviště: {yearMIN}-{yearMAX}"
arcpy.AddMessage(log_all_cropp)
arcpy.SetParameterAsText(18, log_all_cropp)
#####

### PREPARE THE TABLE OF SITES #####
# create empty feature class with sites present in selected records
sr = arcpy.SpatialReference(4326)
arcpy.management.CreateFeatureclass(
    out_path=arcpy.env.scratchGDB,
    out_name="sites_empty",
    geometry_type="POINT",
    spatial_reference = sr)
new_sites = os.path.join(arcpy.env.scratchGDB, "sites_empty")

# add new fields: site_code,site_name,species,latitude, longitude, samples_n, Cm{selected_year}
Cm_fieldname = str('Cm' + selected_year)
fieldnames_new = [('site_code', 'TEXT', 0), ('site_name', 'TEXT', 0), ('species', 'TEXT', 0), ('latitude', 'FLOAT', 7),
('longitude', 'FLOAT', 7), ('samples_n', 'SHORT', 0), (Cm_fieldname, 'FLOAT', 2)] # field_scale used in fact only for float
and double
for fn in fieldnames_new:
    newfield(new_sites, fn[0], fn[1], fn[2])

# add sitecode etc. rows to the feature class + coordinates via 'SHAPE@XY'
with arcpy.da.InsertCursor(new_sites, ["site_code", "SHAPE@XY", "site_name", "species", "latitude", "longitude"]) as
cursor:
    for row in site_packets: # site_packet = (site_code,
(longitude, latitude), site_name, species, latitude, longitude)
        cursor.insertRow(row)

# add fields with data from the rasters
arcpy.sa.ExtractMultiValuesToPoints(
    in_point_features=new_sites,
    in_rasters=[["https://unce-18.natur.cuni.cz/arcgis/rest/services/elevation/ImageServer", "elevation"],
# >>> URL REFERENCE TO RASTER IMAGERY LAYER (ELEVATION) <<<
["https://unce-18.natur.cuni.cz/arcgis/rest/services/slope/ImageServer", "slope"],
# >>> URL REFERENCE TO RASTER IMAGERY LAYER (SLOPE) <<<
["https://unce-18.natur.cuni.cz/arcgis/rest/services/aspect/ImageServer", "aspect"]]
# >>> URL REFERENCE TO RASTER IMAGERY LAYER (ASPECT) <<<
#####

### FILL IN CROPPER VALUES & SAMPLE COUNTS FROM DICTIONARY #####
sites_with_C = []
sites_without_C = []
with arcpy.da.UpdateCursor(new_sites, ["site_code", Cm_fieldname, "samples_n"]) as cursor: # site_code == row[0],
C{selected_year} == row[1], samples_n == row[2]
    for row in cursor:
        for ss_key,ss_value in sites_c_dict.items(): # for each record (sitecode)
            if row[0] == ss_key: # if site code of the row == current site code
                if int(selected_year) in ss_value: # if current year has C. value
                    C_selected_year = ss_value[int(selected_year)]
                    if C_selected_year is not None: sites_with_C.append(ss_key)
                    row[1] = C_selected_year # fill Cropper value median of the selected year to the

```

```

"C{selected_year}" field of the table
    row[2] = sample_counts[row[0]]
    else:
        sites_without_C.append(ss_key)
        cursor.updateRow(row)
if len(sites_without_C) != 0: arcpy.AddWarning(f"Stanoviště {str(sites_without_C).strip('[]')} neobsahují
Cropperovu hodnotu pro rok {selected_year}.\n")
log_sites_to_analysis = f"Počet stanovišť s vypočítaným mediánem Cropperových hodnot v roce {selected_year}:
{len(sites_with_C)}\nStanoviště vstupující do regrese: {str(sites_with_C).strip('[]')}
"
arcpy.AddMessage(log_sites_to_analysis)
arcpy.SetParameterAsText(19, log_sites_to_analysis)

if len(sites_with_C) < 10: # terminate analysis if not enough values for relevance >>> CHANGE HERE THE MINIMAL
NUMBER OF VALUES GOING INTO REGRESSION <<<
    arcpy.AddError(f"Příliš málo stanovišť s mediány Cropperových hodnot za rok {selected_year} pro relevantní
analýzu. Zvolte prosím větší vzorek dat nebo jiný rok.")
    exit()
#####

### PREPARATION FOR ANALYSIS #####
arcpy.management.MakeFeatureLayer("https://unce-18.natur.cuni.cz/arcgis/rest/services/Hosted/prediction_points/
FeatureServer/0", "point_net_lyr") # >>> URL REFERENCE TO VECTOR FEATURE LAYER (PREDICTION POINTS) <<<
#####

### ANALYSIS #####
in_features = new_sites

prediction_locations = "point_net_lyr"

# Generalized Linear Regression
arcpy.stats.GeneralizedLinearRegression(
    in_features=in_features,
    prediction_locations=prediction_locations,
    dependent_variable=Cm_fieldname,
    model_type="CONTINUOUS",
    output_features=os.path.join(arcpy.env.scratchGDB, "regression_output"),
    output_predicted_features=os.path.join(arcpy.env.scratchGDB, "regression_output_predicted"),
    explanatory_variables=explanatory)

log_regression = arcpy.GetMessages()
arcpy.AddMessage(log_regression)
arcpy.SetParameterAsText(20, log_regression)
arcpy.AddMessage("Obecná lineární regrese úspěšně proběhla.")

# Local Polynomial Interpolation
search = arcpy.ga.SearchNeighborhoodStandardCircular(
    radius=0.0045, # approximately equal to 500 meters
    angle=0,
    sectorType="ONE_SECTOR")

arcpy.ga.LocalPolynomialInterpolation(
    in_features=os.path.join(arcpy.env.scratchGDB, "regression_output_predicted"),
    z_field="PREDICTED",
    out_raster=os.path.join(arcpy.env.scratchFolder, "interpol_r"), # >> raster output <<
    cell_size=0.0009, # approximately equal to 100 meters
    power=polynomial,
    search_neighborhood=search,
    kernel_function="GAUSSIAN",
    output_type="PREDICTION")

output_raster_interpolated = os.path.join(arcpy.env.scratchFolder, "interpol_r")
arcpy.SetParameterAsText(22, output_raster_interpolated)

arcpy.AddMessage("Lokální polynomičká interpolace úspěšně proběhla.")
#####

### DEVIATIONS FROM DATAPOINTS #####
# fill in predicted values
pred_name = str('pred' + selected_year)
arcpy.sa.ExtractMultiValuesToPoints(
    in_point_features=new_sites,
    in_rasters=[[os.path.join(arcpy.env.scratchFolder, "interpol_r"), pred_name]])
arcpy.management.CalculateField(
    in_table=new_sites,
    field=pred_name,
    expression=f"round(!{pred_name}!,3)",
    expression_type="PYTHON3")

# calculate & fill in deviations
dev_name = str('dev' + selected_year)
newfield(new_sites, dev_name, 'FLOAT', 2)
difference=f"round(!{Cm_fieldname}!,3)-!{pred_name}!" # PYTHON3 !Cm1997!-!pred1997!
arcpy.management.CalculateField(
    in_table=new_sites,
    field=dev_name,
    expression=difference,
    expression_type="PYTHON3")

```



```

arcpy.AddMessage("Proběhl výpočet odchylek odhadovaných dat od původních dat.")
#####

### RASTER STATISTICS #####
rasterMINObject = arcpy.management.GetRasterProperties(os.path.join(arcpy.env.scratchFolder,"interpol_r"),
"MINIMUM")
rasterMAXObject = arcpy.management.GetRasterProperties(os.path.join(arcpy.env.scratchFolder,"interpol_r"),
"MAXIMUM")
rasterMIN = str(rasterMINObject.getOutput(0)).replace(',','.')
rasterMAX = str(rasterMAXObject.getOutput(0)).replace(',','.')
arcpy.AddMessage(f"Minimální hodnota interpolovaného rasteru: {rasterMIN}")
arcpy.AddMessage(f"Maximální hodnota interpolovaného rasteru: {rasterMAX}")
#####

### RASTER RECLASSIFICATION #####
inRaster = os.path.join(arcpy.env.scratchFolder,"interpol_r")
reclassField="Value"
ranges, interval_step = range_list(float(rasterMIN), float(rasterMAX))
remap = arcpy.sa.RemapRange(ranges)

outReclassify = arcpy.sa.Reclassify(
    in_raster=inRaster,
    reclass_field=reclassField,
    remap=remap,
    missing_values="NODATA")
outReclassify.save(os.path.join(arcpy.env.scratchGDB, "reclassified"))

arcpy.conversion.RasterToPolygon(
    in_raster=os.path.join(arcpy.env.scratchGDB, "reclassified"),
    out_polygon_features=os.path.join(arcpy.env.scratchGDB,"interpolated_polygons"), # >> polygons for rendering
<<
    simplify="NO_SIMPLIFY",
    create_multipart_features="MULTIPLE_OUTER_PART")

output_polygons_reclassified = os.path.join(arcpy.env.scratchGDB,"interpolated_polygons")
arcpy.SetParameterAsText(23,output_polygons_reclassified)

log_interpretation = f"\nInterpretace vykreslených polygonů:\nKaždá třída vyjadřuje interval hodnot (predikovaných
mediánů Cropperových hodnot) o šířce {round(interval_step,3)}.\nTřída 0 vyjadřuje nulové hodnoty, případná třída 6,
resp. -6 vyjadřuje kladné nebo záporné odlehle hodnoty vyšší než {round(5*interval_step,2)}, resp. nižší než {-
round(5*interval_step,2)}.\n"
arcpy.AddMessage(log_interpretation)
arcpy.SetParameterAsText(21, log_interpretation)
#####

### OUTPUTS #####
# output point features
arcpy.management.CopyFeatures(new_sites,os.path.join(arcpy.env.scratchFolder,"sites.shp"))
output_sites_shp = os.path.join(arcpy.env.scratchFolder,"sites.shp")
arcpy.SetParameterAsText(24,output_sites_shp)
#####

### CLEAN UP #####
arcpy.management.Delete(os.path.join(arcpy.env.scratchFolder,"db_all.txt"))
arcpy.management.Delete(os.path.join(arcpy.env.scratchFolder,"db_selection.txt"))
#####

# timer
all_end = time.time()
print(f"Uplynulý čas: {all_end - all_start}")

```

## Příloha 2: Možnost definování symbologie výstupní polygonové vrstvy s využitím JSON rendereru

```
symbology_polygons = """"{
  "type" : "uniqueValue",
  "field1" : "gridcode",
  "field2" : null,
  "field3" : null,
  "fieldDelimiter" : ", ",
  "defaultSymbol" : {
    "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [130,130,130,255],
    "width" : 1
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  },
  "defaultLabel" : "Ostatní hodnoty",
  "uniqueValueInfos" : [
    { "value" : "-6",
      "label" : "Třída -6",
      "description" : "Třída -6",
      "symbol" :
      { "type" : "esriSFS",
        "style" : "esriSFSSolid",
        "color" : [255,0,0,255],
        "outline": {
          "type": "esriSLS",
          "style": "esriSLSSolid",
          "color": [0,0,0,255],
          "width": 1
        }
      }
    },
    { "value" : "-5",
      "label" : "Třída -5",
      "description" : "Třída -5",
      "symbol" :
      { "type" : "esriSFS",
        "style" : "esriSFSSolid",
        "color" : [255,95,0,255],
        "outline": {
          "type": "esriSLS",
          "style": "esriSLSSolid",
          "color": [0,0,0,255],
          "width": 1
        }
      }
    },
    { "value" : "-4",
      "label" : "Třída -4",
      "description" : "Třída -4",
      "symbol" :
      { "type" : "esriSFS",
        "style" : "esriSFSSolid",
        "color" : [255,140,0,255],
        "outline": {
          "type": "esriSLS",
          "style": "esriSLSSolid",
          "color": [0,0,0,255],
          "width": 1
        }
      }
    },
    { "value" : "-3",
      "label" : "Třída -3",
      "description" : "Třída -3",
      "symbol" :
      { "type" : "esriSFS",
        "style" : "esriSFSSolid",
        "color" : [252,177,26,255],
        "outline": {
          "type": "esriSLS",
          "style": "esriSLSSolid",
          "color": [0,0,0,255],
          "width": 1
        }
      }
    },
    { "value" : "-2",
      "label" : "Třída -2",
      "description" : "Třída -2",
      "symbol" :
      { "type" : "esriSFS",
        "style" : "esriSFSSolid",
        "color" : [249,210,72,255],
        "outline": {
          "type": "esriSLS",
          "style": "esriSLSSolid",
          "color": [0,0,0,255],
          "width": 1
        }
      }
    },
    { "value" : "-1",
      "label" : "Třída -1",
      "description" : "Třída -1",
      "symbol" :
      { "type" : "esriSFS",
        "style" : "esriSFSSolid",
        "color" : [241,224,79,255],
        "outline": {
          "type": "esriSLS",
          "style": "esriSLSSolid",
          "color": [0,0,0,255],
          "width": 1
        }
      }
    }
  ]
}"""
```

```

        "width": 1
    }
}
},
{ "value" : "0",
  "label" : "Třída 0",
  "description" : "Třída 0",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [243,248,102,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
{ "value" : "1",
  "label" : "Třída 1",
  "description" : "Třída 1",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [231,238,89,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
{ "value" : "2",
  "label" : "Třída 2",
  "description" : "Třída 2",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [220,251,103,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
{ "value" : "3",
  "label" : "Třída 3",
  "description" : "Třída 3",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [194,252,82,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
{ "value" : "4",
  "label" : "Třída 4",
  "description" : "Třída 4",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [160,253,62,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
{ "value" : "5",
  "label" : "Třída 5",
  "description" : "Třída 5",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [115,254,46,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
{ "value" : "6",
  "label" : "Třída 6",
  "description" : "Třída 6",
  "symbol" :
  { "type" : "esriSFS",
    "style" : "esriSFSSolid",
    "color" : [0,255,35,255],
    "outline": {
      "type": "esriSLS",
      "style": "esriSLSSolid",
      "color": [0,0,0,255],
      "width": 1
    }
  }
},
],
}""
symbology_renderer =
f"JSONRENDERER={symbology_polygons}

```