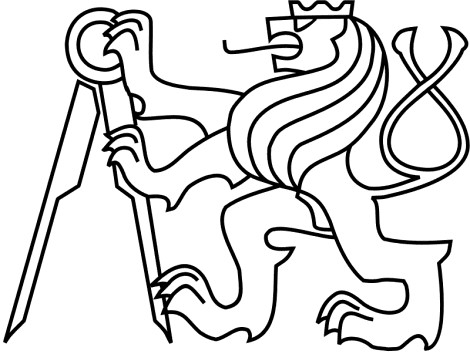
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STAVEBNÍ

STUDIJNÍ PROGRAM GEODÉZIE A KARTOGRAFIE

OBOR GEOMATIKA



**Archeologická naleziště ČR**

155UZPR: Semestrální práce

Bc. Jan Kučera

Bc. Pane Kuzmanov

Bc. Jakub Šimek

Praha, Leden 2021

Obsah

[1 Úvod 3](#_Toc61981625)

[1.1 Zadání 3](#_Toc61981626)

[1.2 Téma a cíl práce 3](#_Toc61981627)

[2 Využitý software 3](#_Toc61981628)

[2.1 QGIS 3](#_Toc61981629)

[2.2 Python 3](#_Toc61981630)

[2.3 VS Code 3](#_Toc61981631)

[2.4 GitHub 4](#_Toc61981632)

[3 Zdrojová data 4](#_Toc61981633)

[3.1 AMČR 4](#_Toc61981634)

[3.2 Další data 4](#_Toc61981635)

[4 Práce s daty 5](#_Toc61981636)

[4.1 Popis dat 5](#_Toc61981637)

[4.1.1 Struktura 5](#_Toc61981638)

[4.2 Python implementace 6](#_Toc61981639)

[4.2.1 main.py 7](#_Toc61981640)

[4.2.2 coinfig.py 7](#_Toc61981641)

[4.3 Stažení a vytvoření tabulek 7](#_Toc61981642)

[4.4 Validace 8](#_Toc61981643)

[4.5 Analýza dat 8](#_Toc61981644)

[4.5.1 SQL dotazy 9](#_Toc61981645)

[4.5.2 Výsledky 9](#_Toc61981646)

[5 Závěr 9](#_Toc61981647)

[6 Zdroje 9](#_Toc61981648)

# Úvod

Tento dokument slouží k dokumentaci semestrální práce na téma Archeologická naleziště z předmětu Úvod do zpracování prostorových dat (155UZPR).

## Zadání

Navrhněte a vytvořte tematické vrstvy (např. vodní toky, vodní plochy, lesy, silnice, železnice a pod.) na základě dat [OpenStreetMap](http://cs.wikipedia.org/wiki/OpenStreetMap) (viz schéma [osm](https://geo.fsv.cvut.cz/gwiki/Cvi%C4%8Dn%C3%A1_datab%C3%A1ze_PostGIS#osm)) a [další otevřených zdrojů](https://data.gov.cz/datov%C3%A9-sady).

Aplikujte testy datové integrity a odstraňte případné nekonzistence v datech.

Vytvořte tutoriál - tj. sadu atributových a prostorových dotazů nad databází uzpr\_projekty.

## Téma a cíl práce

Hlavním tématem této práce je práce s databází projektu Archeologická mapa České republiky a jejich analýza prostřednictvím SQL dotazů. Celkovým cílem je seznámení se s problematikou prostorového SQL dotazování, jak už naznačuje název předmětu, a celkově s manipulací otevřených geodat.

# Využitý software

V této kapitole bude stručně řečeno, které softwary byly během práce využívány a jakým způsobem. Kromě uvedených kapitol byl ještě využíván software Excel společnosti Microsoft pro zobrazení CSV souborů. Jelikož je velice rozšířený a známý, tak je zde pouze zmíněn.

## QGIS

QGIS (Quantum GIS) je jeden z mnoha geoinformačních systému zaměřených na manipulaci s geodaty a je volně dostupný. V tomto projektu bylo hlavně využíváno grafického zobrazování dat a využívání možností prohlížení databází a jejich dotazování jazykem SQL.

## Python

Python je rychle se rozšiřující a stále více využívaný skriptovací programovací jazyk. Pro účely této práce bylo využito jeho vlastností pro automatizaci dotazování a možnosti alternativního přístupu k databázím. Použita byla knihovna *psycopg2*.

## VS Code

Visual Studio Code je vývojové prostředí vyvinuto společností Microsoft. Toto prostředí bylo využito pro vývoj dotazovacího skriptu v jazyce Python.

## GitHub

Je freeware program pro správu verzí. Umožňuje vyvíjet programy, na kterých pracuje vícečlenný tým. Server GitHub hostuje obrovské množství repozitářů. Výhod této platformy v rámci této práce bylo využito prostřednictvím repozitáře UZPR\_archeology.

# Zdrojová data

## AMČR

Hlavními zdrojovými daty byly data z databáze projektu Archeologická mapa České republiky. Jedná se o projekt Archeologického ústavu Akademie Věd České republiky financovaný Ministerstvem kultury ČR v letech 2012-2015. Celkovým cílem tohoto projektu bylo vytvoření interaktivního systému správy informací o archeologickém dědictví na našem území.

Databáze obsahuje popis archeologických nalezišť. Lze zde dohledat kde a co bylo nalezeno. Kdo prováděl výzkum a podobně.

*V rámci této kapitoly bylo čerpáno z [1].*

## Další data

Další data, která byla využita pro analýzu data archeologických nalezišť je databáze infrastruktury INSPIRE. Jedná se o iniciativu Evropské komise pro správu geodat v EU. Odtud byla použita geometrická určení krajů a obcí ČR.

Dalším zdrojem byla data RUIAN, což je Registr územní identifikace, adres a nemovitostí. Jde o jeden z hlavních registrů ČR a je řízen zákonem o základních registrech. Správcem tohoto registru je Český úřad zeměměřický a katastrální (ČUZK). Obsahuje data o základních územních prvcích, z nichž byly využity data správních obvodů v hlavním městě Praha.

AOPK je Agentura ochrany přírody a krajiny ČR. Jedná se o orgán státní správy zajišťující správu chráněných oblastí a přírodních rezervací. Z jejich databáze byla využita data maloplošných chráněných oblastí a velkoplošných chráněných oblastí.

DIBAVOD je Digitální báze vodohospodářských dat a je nadstavbou Základní báze geografických dat (ZABAGED), což je databáze mapující povrch ČR spravovaná úřadem ČUZK. Odtud byla požita data vodních toků. Těch ale bylo až moc pro vykonání prostorového dotazu v reálném čase a tudíž byla namísto nich použita generalizovaná data databáze ArcČR500.

ArcČR500 digitální vektorová geografická databáze ČR, která vznikla ve spolupráci ARCDATA PRAHA, s.r.o., ČUZK a ČSU (Český statistický úřad). Jsou to data generalizovaná pro měřítko 1:500 000. Odtud byly využity řidší data vodních toků a pak dále i data polygonů sídel.

# Práce s daty

Práce byla vykonávána v GitHub repozitáři UZPR\_archeology. Odkaz na stažení této složky je uveden ve zdroji [2].

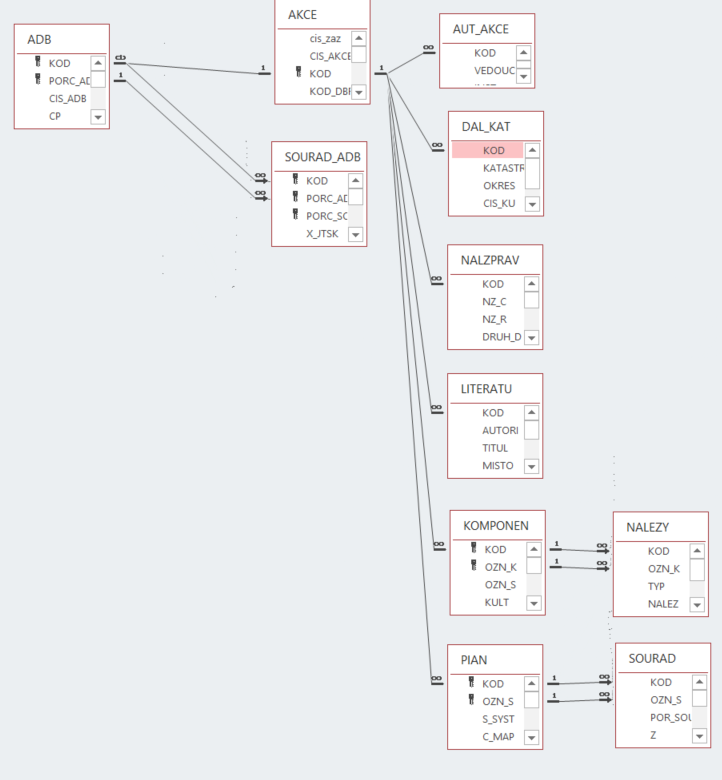
Pro semestrální práci byla využita hlavně souřadnicová data nalezišť v systému S-JTSK v tabulce SOURAD a pak i data z tabulky NALEZY a KOMPONEN pro zjištění předmětu nálezu a odhadované doby jeho vzniku. Detailnější popis bude nastíněn dále.

## Popis dat

Databáze AMČR bohužel obsahuje velice nekonzistentní data, hlavně co se týče nedoplněných informací v některých tabulkách databáze. Polohové určení nalezišť v systému S-JTSK je také problém. Vyskytují se zde špatně označené souřadnice pro souřadnice x a y, což ve výsledku znamená, že při jednotném naimportování je část dat umístěna úplně mimo ČR. K databázi bohužel nebyla nalezena žádná dokumentace, takže se může jednat možná i jen o špatnou interpretaci. Jelikož, ale u zdroje dat nebyl žádný dokument přiložen, byly jsme nuceni vlastní interpretaci provést. Ta bude popsána v dalších kapitolách.

### Struktura

Struktura databáze se sestává celkově ze 17 tabulek, které jsou provázány na základě atributu *kod* a *porc\_adb* pro tabulky s příponou ADB a dále pro ostatní tabulky pouze na základě atributu *kod*. Orientačně jsme se řídili strukturou naznačenou na obrázku 1.



Obrázek 1

Hlavní tabulkou této práce je tabulka SOURAD, která obsahuje pro nás důležitá polohová data v systému S-JTKS.

## Python implementace

Veškeré procesy měly být provedeny pomocí skriptu vytvořeného v jazyce Python prostřednictvím knihovny *psycopg2*. Bohužel editace dat z toho prostředí z nějakého důvodu nefungovala. Posílaní dotazů pro import csv souborů z této knihovny nejevily žádnou zpětnou reakci. Proto v případě importu, dat, vytvoření geometrie dat a prostorového indexu a úpravy datových typů atributů byly dávky posílány přímo z prostředí QGIS a tato část nebyla prostřednictvím Python vykonána. Nicméně napsané metody pro import jsou ve skriptu ponechány pro případ potřeby. Všechny python skripty se nachází ve složce *Python* GitHub repozitáře. Program se sestává ze 2 modulů: *main.py*, *config.py*

### main.py

Toto je hlavní program, ve kterém se nachází všechny metody zprostředkovávající dotazování. Seznam tříd a jejich metod s popisem je následovný:

* třída *Files*:
  + *path\_leaf* – extrakce jména souboru (bez přípony)
* třída *DB*:
  + *send\_query* – posílá dávky a může vracet výstup
  + *create\_table* – vytváří databázovou tabulku (voláno metodou *import\_csv*)
  + *create\_atributes* – vrací string pro vytvoření struktury tabulky (voláno metodou *create\_table*)
  + *import\_csv* – importuje csv soubory do databáze
  + *get\_buffer\_count* – pošle dávku pro zjištění počtu prvků v buffer zóně
  + *get\_bufferZones\_count* – pošle dávku pro zjištění histogramu prvků v pravidelných buffer zónách a vypíše výsledek
  + *get\_atribute\_histogram* – pošle dávku a vypíše histogram počítaného atributu
  + *get\_area\_count* – spočítá počet prvků v zadané oblasti a vypíše výsledek
  + *get\_intersected\_area\_count* - spočítá počet prvků v průniku 2 zadaných oblastí a vypíše výsledek
* funkce *main* – obsahuje zakomentovaná volání metod a pak dále seznam dalších dávek,

Metody *import\_csv*, *create\_table*, *create\_atributes* nejsou plně funkční kvůli problematice odezvy databáze. Při poslání sql dávky pro vytvoření tabulky se totiž vůbec nic nestane.

Celý chod pak zajišťuje funkce *main*, která obsahuje všechny dávky ať už zřízené metodou pro třídu *DB* nebo dávky, které byly poslány rovnou metodou *send\_query* a nebyly pro ně vytvořeny zvláštní metody.

Celý modul a jeho jednotlivé metody jsou kompletně popsány (v angličtině) uvnitř. Pro více informací o jednotlivých modulech je možno nahlédnout přímo do skriptu.

### coinfig.py

Je modul obsahující informace pro přihlášení do databáze *uzpr\_projekty*, úroveň logování loggeru z knihovny logging, cesty csv souborů s jejich názvy a další. Byl vytvořen pro kolekci těchto absolutních hodnot za účelem vytvoření přehlednosti. Prvky toho modulu jsou volány v metodách modulu *main.py*.

## Stažení a vytvoření tabulek

Data byla stažena do adresáře *data*/*tabulky* v repozitáři *UZPR\_archeology* platformy GitHub. Zdroj stažení je uveden ve zdroji [2]. Jedná se o soubory formátu CSV (Comma Separated Values), které budou importovány do databáze *uzpr21\_b* ve schématu *uzpr\_projekt*.

Import do databáze těchto CSV souborů byl realizován v prostředí QGIS databázového manageru. Jedná se o soubory *SOURAD.csv, KOMPONEN.csv* a *NALEZY.csv* (pro názvy souborů byl užita pouze malá písmena). Soubory bez souřadnic byly naimportovány pouze jako atributové tabulky bez geometrie. Pro soubor *SOURAD.*csv byly po importu poslány dávky pro vytvoření geometrie. Nejdříve byl založen atribut nosící geometrii:

SELECT AddGeometryColumn ('sourad','geom',5514,'POINT',2);

Poté byly datové typy atributů obsahující souřadnice převedeny na číselný typ double precision:

ALTER TABLE uzpr21\_b.sourad

    ALTER x\_jtsk TYPE double precision USING x\_jtsk::double precision;

ALTER TABLE uzpr21\_b.sourad

    ALTER y\_jtsk TYPE double precision USING y\_jtsk::double precision;

Nakonec byly vytvořeny geometrie prvků jako body o zadaných S-JTSK souřadnicích prostřednictvím editace již vytvořeného atributu geom:

UPDATE sourad SET geom = ST\_SetSRID(ST\_MakePoint(-y\_jtsk, -x\_jtsk), 5514);

Takto vytvořené geometrie bodů ze zadaných dat leží na území ČR.

## Validace

Protože geometrie dat byly pouze body, mohly se zde projevit datové neintegrity pouze ve formě duplicitních bodu. Pro zjištění, zda jsou data validní byla poslána následující dávka:

SELECT st\_isvalid(sourad.geom)

                FROM sourad

                WHERE st\_isvalid(sourad.geom) IS NOT TRUE;

Výstup nebyl žádný, tedy vstupem byla pouze validní data.

## Analýza dat

Teď, když jsou k dispozici validní data je možné provést na datech některé analýzy. Jednalo se o dávky zjišťující, kde se převážně naleziště vyskytují. Byla testována území podél řek, kolem sídel, v chráněných oblastech pod správou AOPK. Pak byly tvořeny histogramy podle krajů a obcí a také histogram samotných nalezených objektů.

### SQL dotazy

Některé dotazy byly formovány pomocí metod knihovny *DB* v Python skriptu *main.py*, pro zaručení automatizace. Některé dávky zase byly poslány předáním celého dotazu v podobě textového řetězce metodě *send\_query*. U některých jsou uvedené u grafické výsledky vytvořené v programu Matlab pro názornost. Zde jsou uvedené použité dávky.

#### Histogram předmětů nalezišť

Provedeno následující metodou:

database = DB()

database.get\_atribute\_histogram(UZPR\_PROJEKT, "nalezy", "specif", 10)

Dávka by po přeložení vypadala takto:

SELECT specif, count(specif)

FROM nalezy

GROUP by specif

HAVING count(specif) > 10

ORDER BY count(specif) DESC;

Výsledek:

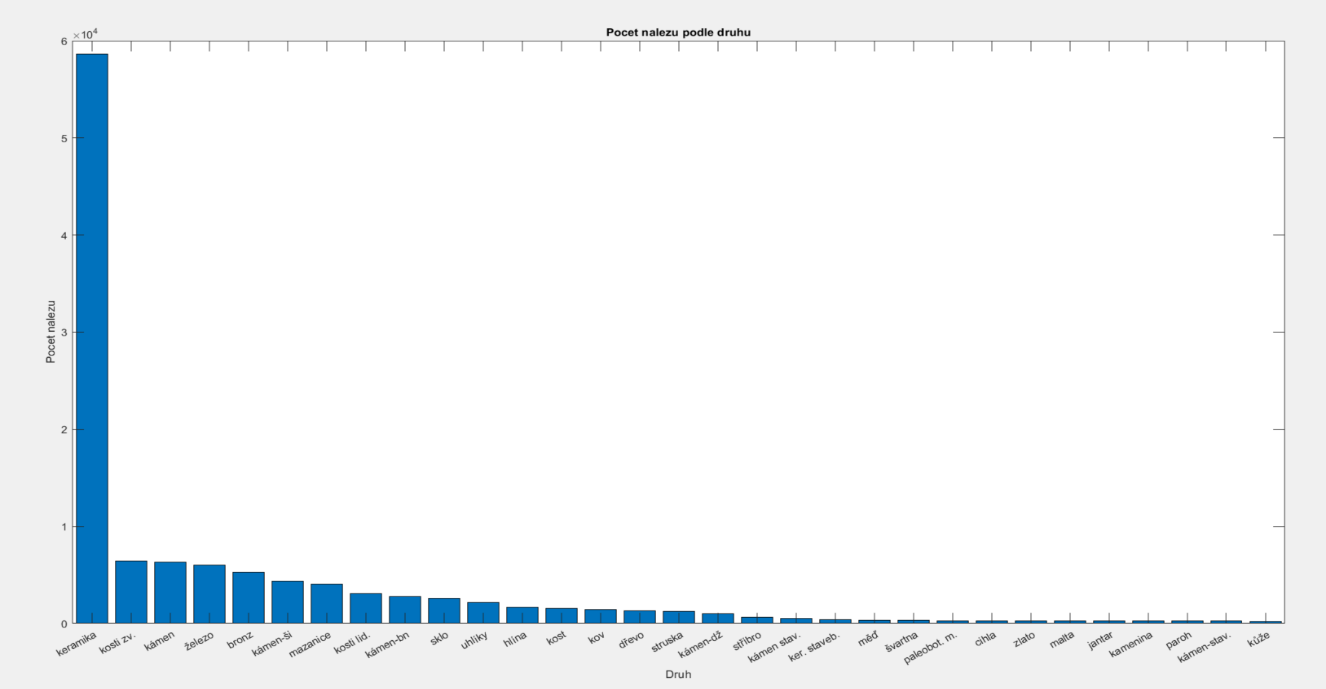
keramika: 58657

kosti zv.: 6452

kámen: 6328

železo: 6014

…



Obrázek 2- Histogram nalezených předmětů

#### Histogram nalezišť v buffer zónách kolem řek

Provedeno následující metodou:

database = DB()

database.get\_bufferZones\_count(UZPR\_PROJEKT, 100, 10, "sourad", "vodnitoky")

Dávka by po přeložení pro každou zónu vypadala takto (buffer se mění pro krok 100m):

SELECT count(\*)

FROM   sourad as targetFeautre

JOIN   vodnitoky as bufferFeautre

ON     targetFeautre.geom @ bufferFeautre.geom

AND    st\_within(targetFeautre.geom, st\_buffer(bufferFeautre.geom, 100));

Výsledek:

Current buffer zone: [ 0m - 100m]; Amout of archeology spots: 21697

Current buffer zone: [100m - 200m]; Amout of archeology spots: 24375

Current buffer zone: [200m - 300m]; Amout of archeology spots: 18982

Current buffer zone: [300m - 400m]; Amout of archeology spots: 15153

Current buffer zone: [400m - 500m]; Amout of archeology spots: 10554

Current buffer zone: [500m - 600m]; Amout of archeology spots: 8488

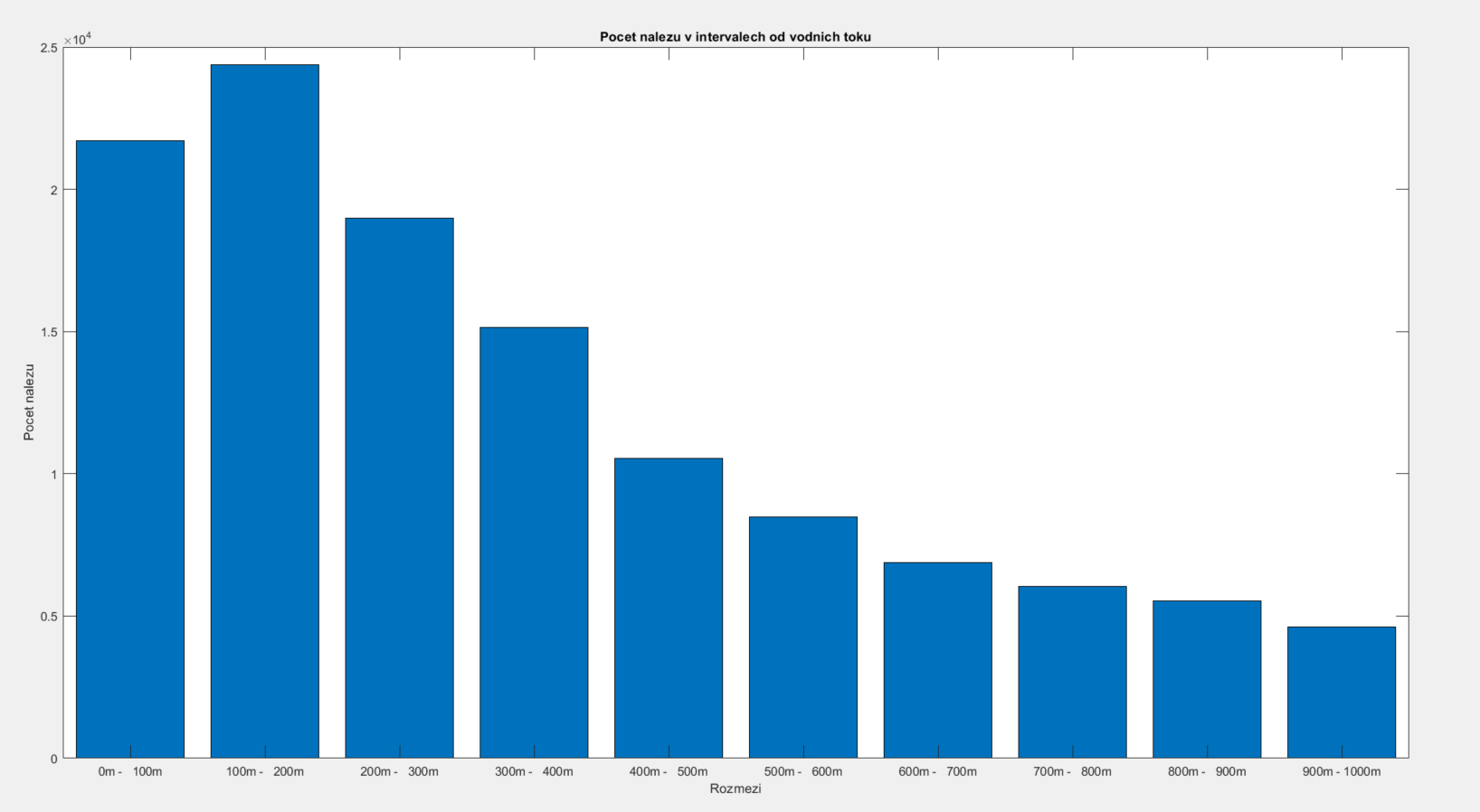
Current buffer zone: [600m - 700m]; Amout of archeology spots: 6890

Current buffer zone: [700m - 800m]; Amout of archeology spots: 6039

Current buffer zone: [800m - 900m]; Amout of archeology spots: 5544

Current buffer zone: [900m - 1000m]; Amout of archeology spots: 4620

Total: 122342



Obrázek 3- Histogram nalezišť v zónách kolem vodních toků

#### Histogram nalezišť v buffer zónách kolem sídel

Provedeno následující metodou:

database = DB()

database.get\_bufferZones\_count(UZPR\_PROJEKT, 100, 10, "sourad", "sidlaplochy")

Dávka by po přeložení vypadala takto:

SELECT count(\*)

FROM   sourad as targetFeautre

JOIN   sidlaplochy as bufferFeautre

ON     targetFeautre.geom @ bufferFeautre.geom

AND    st\_within(targetFeautre.geom, st\_buffer(bufferFeautre.geom, 100));

Výsledek:

Current buffer zone: [0m - 100m]; Amout of archeology spots: 43003

Current buffer zone: [100m - 200m]; Amout of archeology spots: 1559

Current buffer zone: [200m - 300m]; Amout of archeology spots: 1043

Current buffer zone: [300m - 400m]; Amout of archeology spots: 736

Current buffer zone: [400m - 500m]; Amout of archeology spots: 465

Current buffer zone: [500m - 600m]; Amout of archeology spots: 387

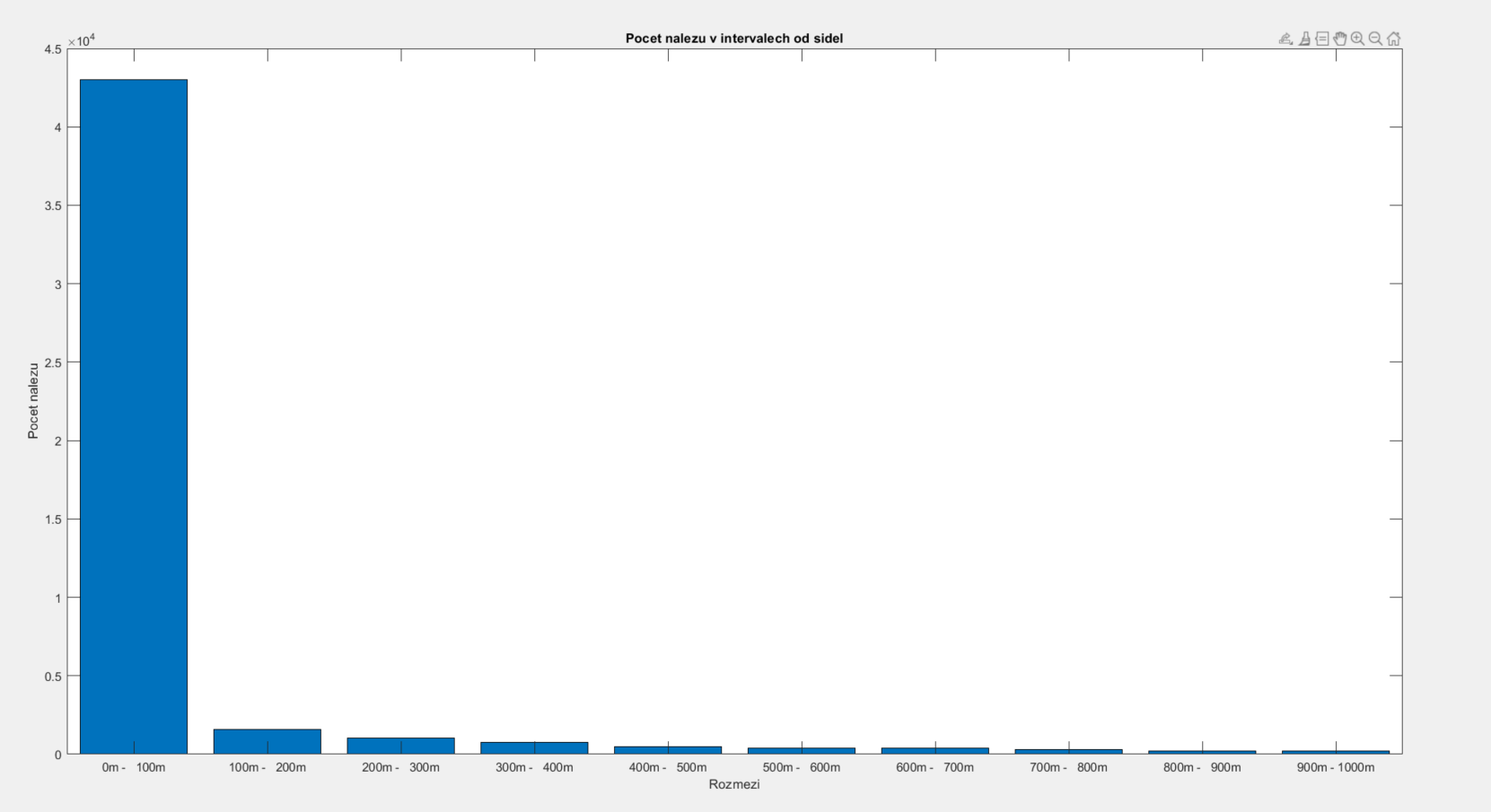
Current buffer zone: [600m - 700m]; Amout of archeology spots: 391

Current buffer zone: [700m - 800m]; Amout of archeology spots: 277

Current buffer zone: [800m - 900m]; Amout of archeology spots: 214

Current buffer zone: [900m - 1000m]; Amout of archeology spots: 205

Total spots: 48280



Obrázek 4- Histogram nalezišť v zónách kolem sídel

#### Počet nalezišť v maloplošných chráněných oblastech

Provedeno následující metodou:

database = DB()

database.get\_area\_count(UZPR\_PROJEKT, "sourad", "aopk.maloplosna\_chranena\_uzemi")

Dávka by po přeložení vypadala takto:

SELECT count(\*)

FROM sourad as s

JOIN aopk.maloplosna\_chranena\_uzemi as u

ON s.geom @ u.geom

AND st\_within(s.geom, u.geom);

**Výsledek**:

2334

#### Počet nalezišť ve velkoplošných chráněných oblastech

Provedeno následující metodou:

database = DB()

database.get\_area\_count(UZPR\_PROJEKT, "sourad", "aopk.velkoplosna\_chranena\_uzemi")

Dávka by po přeložení vypadala takto:

SELECT count(\*)

FROM sourad as s

JOIN aopk.velkoplosna\_chranena\_uzemi as u

ON s.geom @ u.geom

AND st\_within(s.geom, u.geom);

**Výsledek**:

13471

#### Počet nalezišť ve velkoplošných a maloplošných oblastech zároveň

Provedeno následující metodou:

database = DB()

database.get\_intersected\_area\_count(UZPR\_PROJEKT, "sourad", "aopk.maloplosna\_chranena\_uzemi","aopk.velkoplosna\_chranena\_uzemi")

Dávka by po přeložení vypadala takto:

SELECT count(\*)

FROM (

      SELECT s.geom

      FROM sourad as s

      JOIN aopk.maloplosna\_chranena\_uzemi as u

      ON s.geom @ u.geom

      AND st\_within(s.geom, u.geom)

) as s

JOIN aopk.velkoplosna\_chranena\_uzemi as u

ON s.geom @ u.geom

AND st\_within(s.geom, u.geom)

**Výsledek**:

820

#### Histogram nalezišť podle krajů

database = DB()

query = """ SELECT k.text, count(\*)

                FROM sourad as s

                FULL JOIN inspire\_au as k

                ON s.geom @ k.geom

                AND st\_within(s.geom, k.geom)

                WHERE k.localisedcharacterstring = 'Kraj'

                GROUP BY k.text;"""

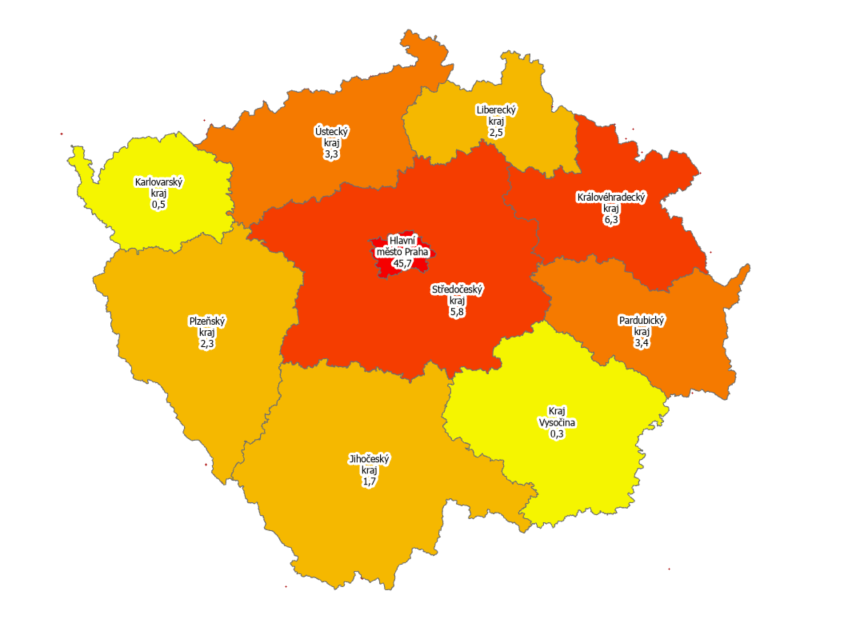
database.send\_query(UZPR\_PROJEKT, query)

**Výsledek**:

Hlavní město Praha 22673

Jihočeský kraj 17367

…



Obrázek 5- Kartogram nalezišť krajů podle poměru počtu nálezů na km2

#### Histogram nalezišť podle obcí

database = DB()

query = """ SELECT k.text, count(\*)

                FROM sourad as s

                FULL JOIN inspire\_au as k

                ON s.geom @ k.geom

                AND st\_within(s.geom, k.geom)

                WHERE k.localisedcharacterstring = 'Obec'

                GROUP BY k.text

                ORDER BY count(s.geom) DESC

                LIMIT 20;"""

database.send\_query(UZPR\_PROJEKT, query)

**Výsledek**:

Praha 22673

Plzeň 2399

…

#### Histogram nalezišť podle časových epoch, odkdy nálezy pocházejí

database = DB()

query = """ SELECT field\_2, count(\*)

                FROM komponen as k

                JOIN doby1 as d

                ON k.kult =  d.kult

                GROUP BY field\_2

                ORDER BY count(\*) DESC;"""

database.send\_query(UZPR\_PROJEKT, query)

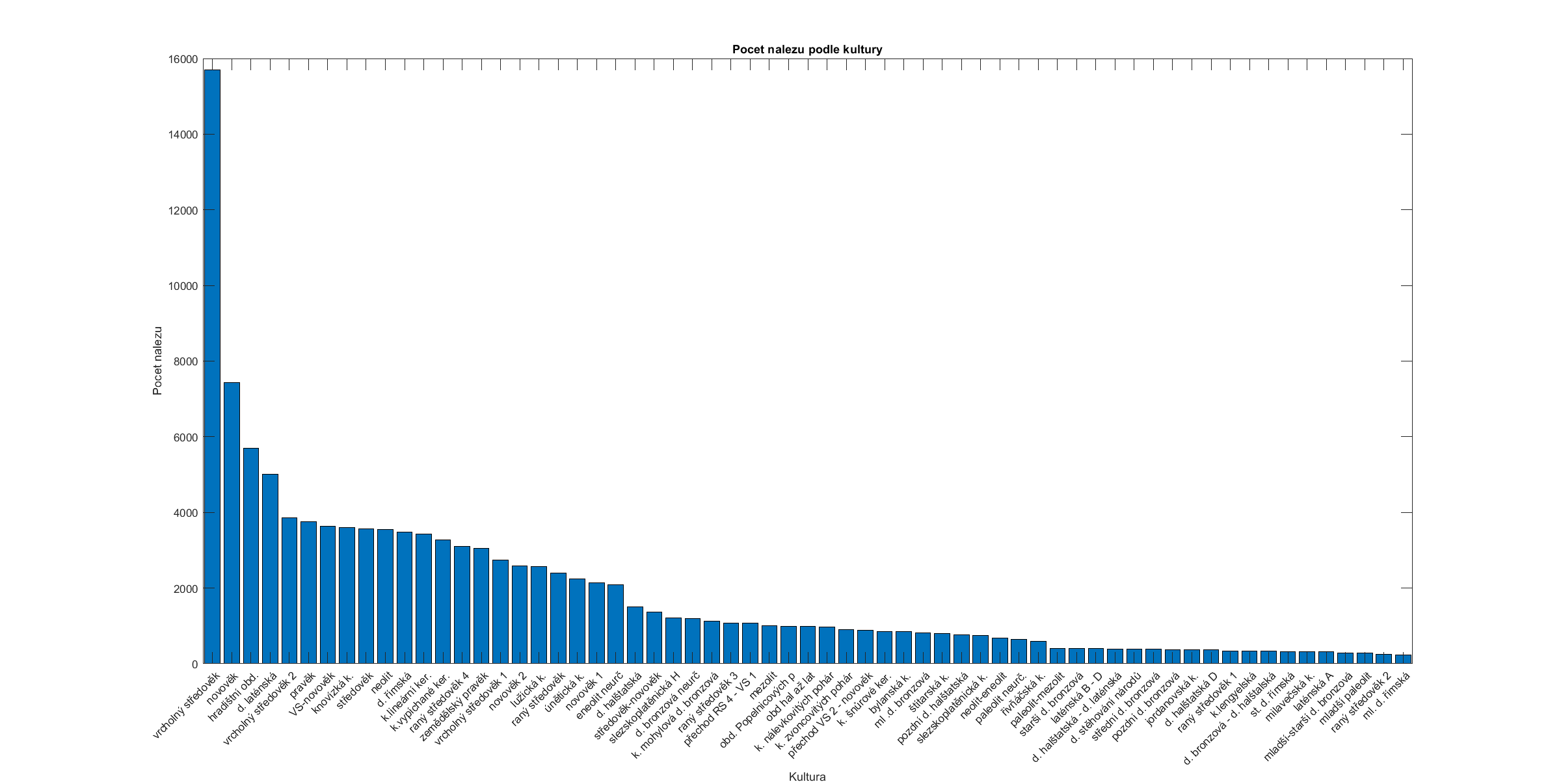
**Výsledek**:

vrcholný středověk 15698

novověk 7432

hradištní obd. 5693

…



Obrázek 6- Histogram nalezišť podle kultur

#### Histogram nalezišť podle správních obvodů v Praze

database = DB()

query = """ SELECT o.nazev, count(\*)

                FROM sourad  as s

                JOIN ruian\_praha.spravniobvody as o

                ON s.geom @ o.geom

                AND st\_within(s.geom, o.geom)

                GROUP BY o.nazev

                ORDER BY count(\*) DESC;"""

database.send\_query(UZPR\_PROJEKT, query)

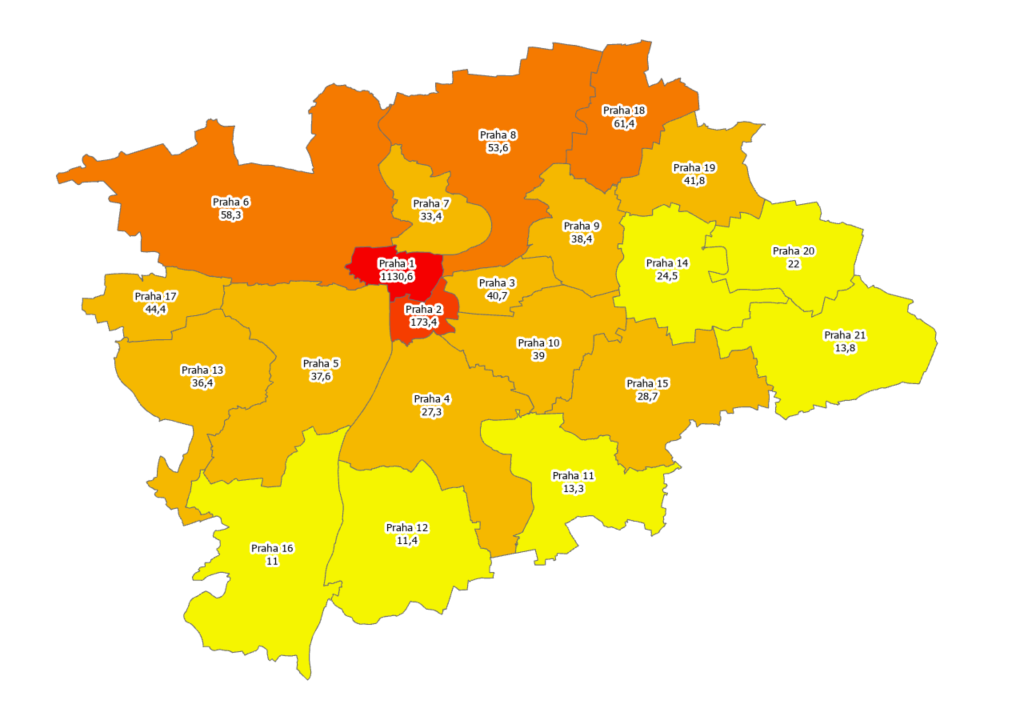
**Výsledek**:

Praha 1 6263

Praha 6 3273

Praha 8 2013

…



Obrázek 7- Kartogram nalezišť podle poměrů počtu nalezišť na km2

# Závěr

Byla vytvořena databáze pro vybraná data AMČR. Byla vytvořena geometrie dat a provedena jejich validace. Poté byly vytvořeny SQL dávky na pro analýzu těchto dat společně s daty z jiných zdrojů.

Úloha byla provedena pomocí python skriptu. Vyskytly se zde potíže s importem dat, proto tato část musela být provedena přímo v programu QGIS. Ostatní dotazování už bylo provedeno skriptem.

V datech AMČR jsou chyby v uvedených S-JTSK souřadnicích, které by asi měli projít obecnou kontrolou. Co se týče výstupu analýz, tak se dá konstatovat, že data jsou urbanisticky rozložena a nejvíce se jich vyskytuje ve městě Praha, jakožto historického a kulturního srdce České republiky. Též se většina nalezišť nachází v okolí řek, kde se odjakživa zakládaly první osady a proto jsou významnější pro výskyt archeologických předmětů.

# Zdroje

[1] Archeologická mapa České republiky [online] [cit. 2021-19-01]. Dostupné z: <http://www.archeologickamapa.cz/?page=project>

[2] GitHub UZPR\_archeology [online] Dostupné z: [https://github.com/Bambojooo/UZPR\_archeology.git](%20https://github.com/Bambojooo/UZPR_archeology.git)

[3] Geoportál ČUZK [online]. Dostupné z: <https://geoportal.cuzk.cz/(S(xkxbqq0ef1f1n1snbw3irlnu))/Default.aspx?mode=TextMeta&side=dSady_RUIAN&metadataID=CZ-00025712-CUZK_SERIES-MD_AU&menu=334>