Andrzej Dudek Eugeniusz Gatnar Dominik Rozkrut Marek Walesiak

Pozyskiwanie danych z API Banku Danych Lokalnych w środowisku R

Andrzej.Dudek@ue.wroc.pl
Eugeniusz.Gatnar@ue.katowice.pl
D.Rozkrut@stat.gov.pl
Marek.Walesiak@ue.wroc.pl

Plan prezentacji

1. Cel prezentacji
2. Bank Danych Lokalnych
3. Pakiet bdl oraz interfejs API
4. Procedury automatycznego pozyskiwania danych
   – szeregi przekrojowe
   – szeregi czasowe
   – szeregi czasowo-przekrojowe (panelowe)
5. Zastosowanie z wykorzystaniem drzew regresyjnych
6. Podsumowanie

1. Cel prezentacji

W referacie zaprezentowano:

1. Nowy sposób automatycznego pozyskiwania danych z Banku Danych Lokalnych z wykorzystaniem pakietu bdl oraz interfejsu API (Application Programming Interface)
2. Architekturę interakcji BDL<->API<->pakiet bdl <-> program R
3. Procedury automatycznego pozyskiwania danych dla szeregów przekrojowych, czasowych, oraz czasowo-przekrojowych
4. Przykład z wykorzystaniem drzew regresyjnych

2. Bank Danych Lokalnych – rys historyczny

sierpień 1993	**Zarządzenie wewnętrzne nr 13 Prezesa GUS: „WUS w Jeleniej Górze pełni rolę wiodącą w zakresie prac związanych z budową systemu statystyki lokalnej w GUS, w tym utworzenia Banku Danych Lokalnych (BDL)”**
marzec 1995	wskazanie **Ośrodka Informatyki Statystycznej (OIS) w Radomiu** jako partnera informatycznego projektu
marzec 1998	**BDL dostępny jest w Internecie**
2001-2003	zmiana systemu bazodanowego z Ingres na **Oracle**
2014-2017	zmiana systemu bazodanowego **Oracle** na **Microsoft SQL Server**
2018	**BDL API (4 XII 2018 R.)**

2. Bank Danych Lokalnych – siedziby głównych twórców

2. Bank Danych Lokalnych – charakterystyka

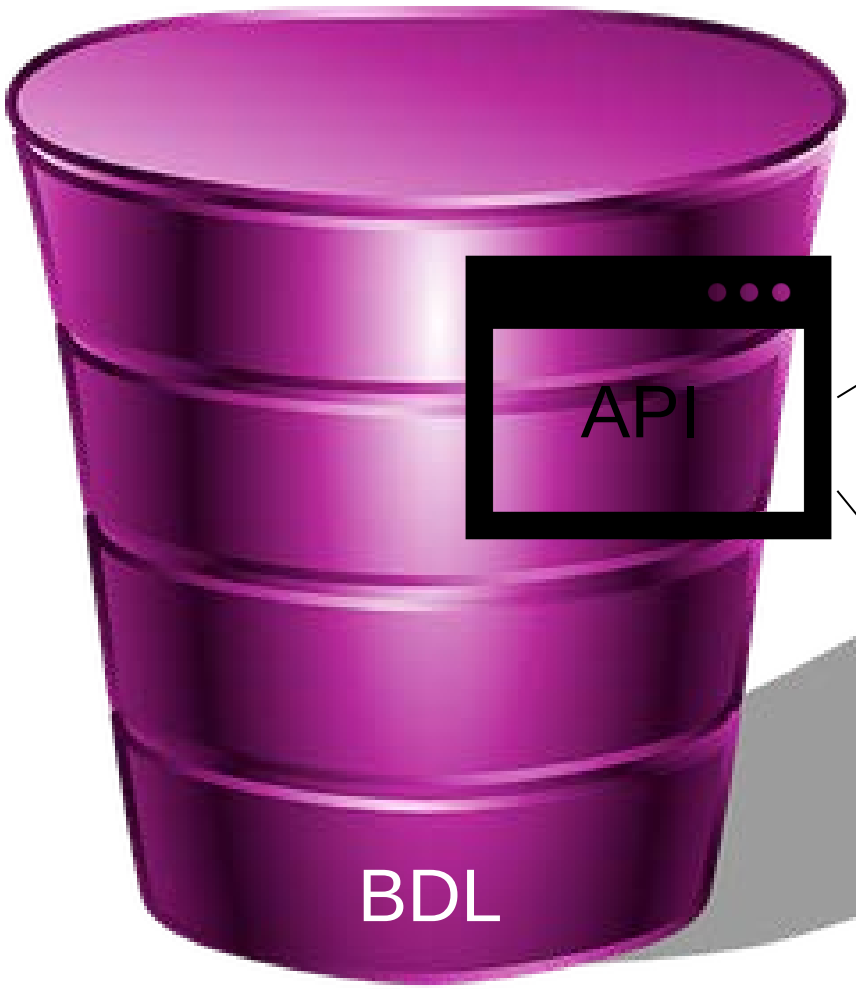
- strona **https://bdl.stat.gov.pl**
- jest największym w Polsce uporządkowanym zbiorem informacji o sytuacji gospodarczej, demograficznej, społecznej oraz stanie środowiska
- oferuje ponad 96 tys. cech statystycznych (każda ma indywidualny kod ID) pogrupowanych tematycznie
- obejmuje 32 dziedziny tematyczne
- poziom agregacji danych obejmuje układ administracyjny (kraj, województwa, powiaty, gminy, miejscowości) oraz NUTS (makroregiony – NUTS1, regiony – NUTS2, podregiony – NUTS3)
- dane prezentowane są w ujęciu rocznym (od 1995) i krótkookresowym (od 2005)

Pakiet bdl jest udostępniony obecnie na stronie

https://github.com/KaniaKrzysztof/bdl

Lp.	Kategoria (dziedzina tematyczna)	Liczba cech
1	Ceny	4069
2	Finanse przedsiębiorstw (dane kwartalne)	6160
3	Finanse publiczne	2563
4	Fundusze unijne (dane półroczne)	26098
5	Gospodarka mieszkaniowa i komunalna	715
6	Handel i gastronomia	84
7	Inwestycje i środki trwałe	397
8	Kultura fizyczna, sport i rekreacja	138
9	Kultura i sztuka	475
10	Ludność	7782
11	Narodowe spisy powszechne	9065
12	Nauka i technika	491
13	Ochrona zdrowia, opieka społeczna i świadczenia na rzecz rodziny	1053
14	Organizacja państwa i wymiar sprawiedliwości	145
15	Podmioty gospodarki narodowej, przekształcenia własnościowe i strukturalne	4835
16	Podział terytorialny	84
17	Powszechne spisy rolne	4924
18	Przemysł i budownictwo	6246
19	Rachunki regionalne	316
20	Rolnictwo, leśnictwo i łowiectwo	1199
21	Rynek materiałowy i paliwowo-energetyczny	57
22	Rynek nieruchomości	970
23	Rynek pracy	4917
24	Samorząd terytorialny	433
25	Sektor non-profit	166
26	Stan i ochrona środowiska	728
27	Szkolnictwo	6557
28	Szkolnictwo wyższe	3360
29	Transport i łączność	787
30	Turystyka	1010
31	Wychowanie przedszkolne	389
32	Wynagrodzenia i świadczenia społeczne	379
	RAZEM	96592

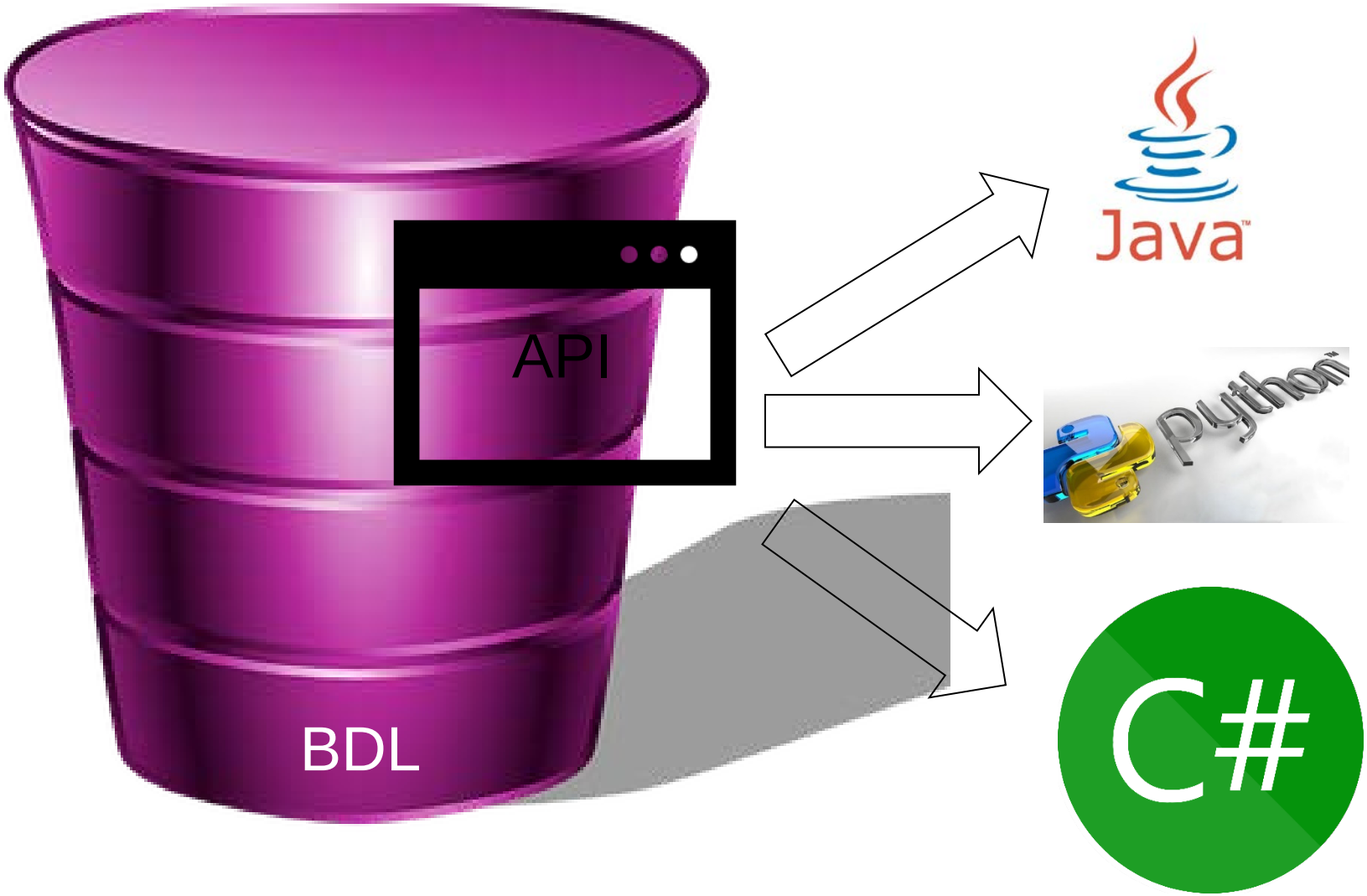
3. Pakiet bdl oraz interfejs API



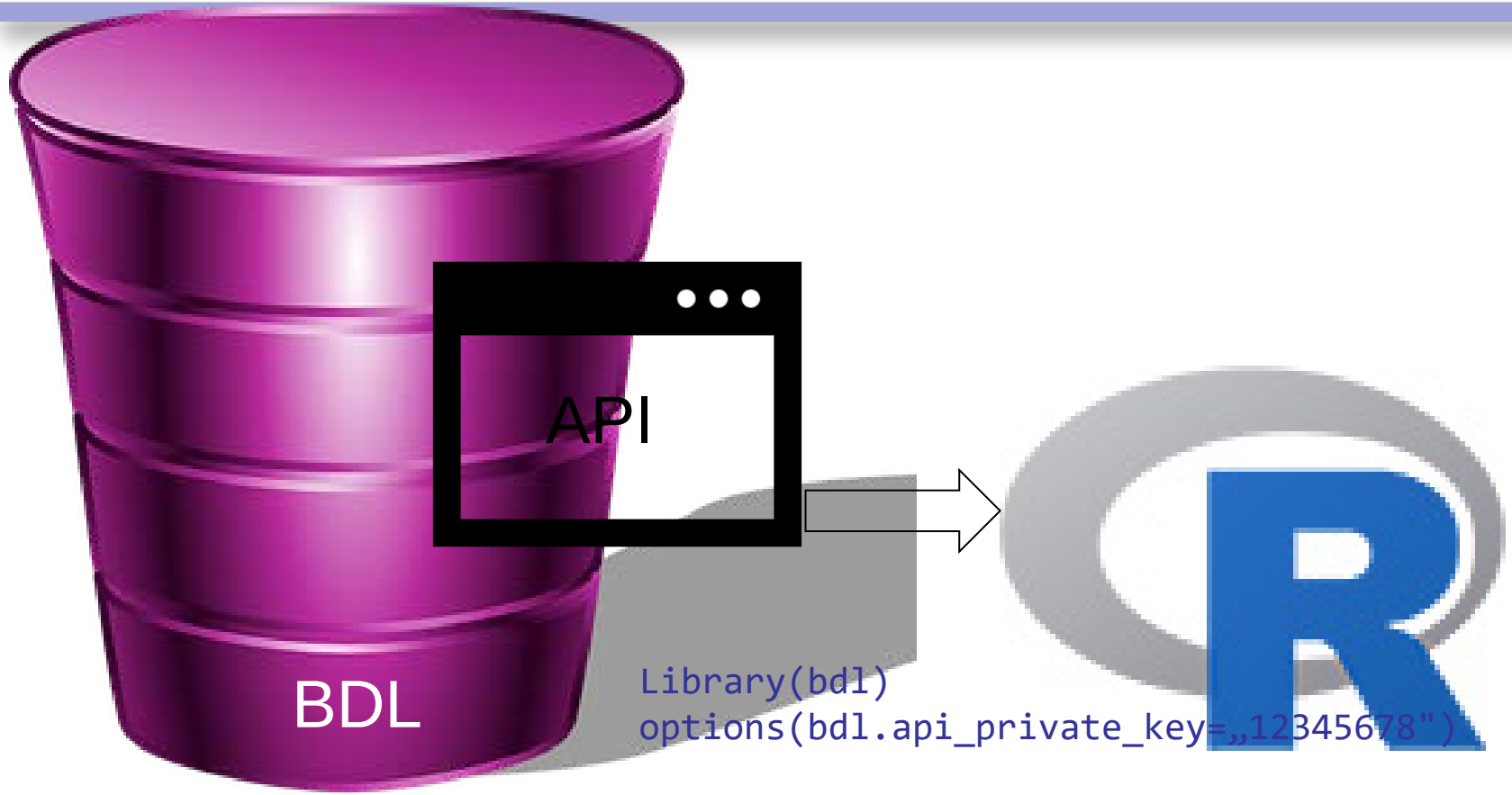

```
unitId:        "000000000000"
unitName:      "POLSKA"
values:
  0:
      year:    "2000"
      val:     75
      attrId:  1
  1:
      year:    "2010"
      val:     78
      attrId:  1
```

```
<unitData>
    <unitId>000000000000</unitId>
    <unitName>POLSKA</unitName>
  - <values>
    - <yearVal>
          <year>2000</year>
          <val>75</val>
          <attrId>1</attrId>
      </yearVal>
    - <yearVal>
          <year>2010</year>
          <val>78</val>
          <attrId>1</attrId>
      </yearVal>
    </values>
</unitData>
```

3. Pakiet bdl oraz interfejs API



3. Pakiet bdl oraz interfejs API




```
Library(bdl)
options(bdl.api_private_key=„12345678")

lp <- get_eurostat_geospatial(output_class = "sf",
resolution = "60", nuts_level = "1")

lp=lp %>% filter (startsWith(NUTS_ID,"PL"))
z=lp %>%  select(NUTS_ID)
data<-get_data_by_variable(varId = 33507,unitLevel
= 2,year = 2017)
```

4. Procedury automatycznego pozyskiwania danych – szeregi przekrojowe

Ocena poziomu atrakcyjności turystycznej powiatów województwa dolnośląskiego w roku 2017:

v1 – miejsca noclegowe na 1000 ludności powiatu,

v2 – udzielone noclegi na 1000 ludności powiatu,

v3 – udzielone noclegi turystom zagranicznym (nierezydentom) na 1000 ludności powiatu,

v4 – emisja zanieczyszczeń pyłowych w tonach na 10 km^2 powierzchni powiatu,

v5 – emisja zanieczyszczeń gazowych w tonach na 1 km^2 powierzchni powiatu,

v6 – przestępstwa o charakterze kryminalnym, przeciwko życiu i zdrowiu oraz przeciwko mieniu na 1000 ludności powiatu,

v7 – lesistość powiatu w %

v8 – uczestnicy imprez masowych na 1000 ludności powiatu,

v9 – podmioty gospodarki turystycznej (sekcje: I, N79) zarejestrowane w systemie REGON na 1000 ludności powiatu.

4. Procedury automatycznego pozyskiwania danych – szeregi przekrojowe

Zmienna dostępne w BDL	ID	Wymagane przeliczenia
x1 – miejsca noclegowe na 1000 ludności powiatu	60300	v1 = x1 (nie wymaga)
x2 – udzielone noclegi na 1000 ludności powiatu	60298	v2 = x2 (nie wymaga)
x3 – udzielone noclegi turystom zagranicznym (nierezydentom)	148102	v3 = (x3 / x6) * 1000
x4 – emisja zanieczyszczeń pyłowych w tonach na 1 km^2 powierzchni powiatu	458168	v4 = x4 * 10
x5 – emisja zanieczyszczeń gazowych w tonach	2090	v5 = x5 / x7
x6 – liczba ludności	72305	X
x7 – powierzchnia powiatu w km^2	2018	X
x8 –Powierzchnia powiatu w ha	1	X
x9 – przestępstwa stwierdzone przez Policję o charakterze kryminalnym na 1000 mieszkańców	498627	v6 = x9+x10+x11
x10 – przestępstwa stwierdzone przez Policję przeciwko życiu i zdrowiu na 1000 mieszkańców	498624	X
x11 – przestępstwa stwierdzone przez Policję przeciwko mieniu na 1000 mieszkańców	498623	X
x12 –Powierzchnia lasów w ha	217916	v7 = (x12 / x8) * 100
x13 – uczestnicy imprez masowych	410728	v8 = (x13 / x6) * 1000
x14 – podmioty gospodarki turystycznej (sekcja I) zarejestrowane w systemie REGON	265296	v9 = ((x14+x15) / x6)*1000
x15 – podmioty gospodarki turystycznej (sekcja N79) zarejestrowane w systemie REGON	266312	X

4. Procedury automatycznego pozyskiwania danych – szeregi przekrojowe

```
library(bdl); library(tidyr); library(magrittr); library(dplyr)
# Indywidualny klucz BDL
options(bdl.api_private_key="????????????????????????????????")
# stałe
rok<-2017
no_uL=5 # (0 - Polska, 1 - makroregion, 2 - województwo, 3 - region, 4 - podregion, 5 - powiat, 6 - gmina, 7 - miejscowość)
# Wczytanie zbioru województw
z<-as.data.frame(get_units(level=2))
# Województwo dolnośląskie
woj=z %>% dplyr::filter(name=="DOLNOŚLĄSKIE")
# Wczytanie informacji o zmiennych z BDL
d1<-get_data_by_variable("60300",unitParentId=woj[,'id'], unitLevel=no_uL,year=rok)
d2<-get_data_by_variable("60298",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d3<-get_data_by_variable("148102",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d4<-get_data_by_variable("458168",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d5<-get_data_by_variable("2090",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d6<-get_data_by_variable("72305",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d7<-get_data_by_variable("2018",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d8<-get_data_by_variable("1",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d9<-get_data_by_variable("498627",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d10<-get_data_by_variable("498624",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d11<-get_data_by_variable("498623",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d12<-get_data_by_variable("217916",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d13<-get_data_by_variable("410728",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d14<-get_data_by_variable("265296",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
d15<-get_data_by_variable("266312",unitParentId=woj[,'id'],unitLevel=no_uL,year=rok)
```

4. Procedury automatycznego pozyskiwania danych – szeregi przekrojowe

```
# Połączenie zmiennnych w jedną ramkę danych
data <- d1 %>%
  full_join (d2, by=c("id","name")) %>%
  full_join (d3, by=c("id","name")) %>%
  full_join (d4, by=c("id","name")) %>%
  full_join (d5, by=c("id","name")) %>%
  full_join (d6, by=c("id","name")) %>%
  full_join (d7, by=c("id","name")) %>%
  full_join (d8, by=c("id","name")) %>%
  full_join (d9, by=c("id","name")) %>%
  full_join (d10, by=c("id","name")) %>%
  full_join (d11, by=c("id","name")) %>%
  full_join (d12, by=c("id","name")) %>%
  full_join (d13, by=c("id","name")) %>%
  full_join (d14, by=c("id","name")) %>%
  full_join (d15, by=c("id","name"))
# Usunięcie z nazwy obiektu wyrazu Powiat
data<-mutate(data,name=gsub("Powiat ","",name))
x<-as.data.frame(data)
# Nadanie nazw wierszom i kolumnom
nazwy<-x[,2]
rownames(x)<-nazwy
x<-x[,c(4,7,10,13,16,19,22,25,28,31,34, 37,40,43,46)]
colnames(x)<-c("x1","x2","x3","x4","x5","x6","x7",
   "x8","x9","x10","x11","x12","x13","x14","x15")
```

```
# Przeliczenia
x<-mutate(x,x3=round((x3/x6)*1000,3))
x<-mutate(x,x4=x4*10)
x<-mutate(x,x5=round(x5/x7,3))
x<-mutate(x,x9=x9+x10+x11)
x<-mutate(x,x12=(x12/x8)*100)
x<-mutate(x,x13=(x13/x6)*1000)
x<-mutate(x,x14=((x14+x15)/x6)*1000)
x<-select(x,-c(x6,x7,x8,x10,x11,x15))
# Nadanie nazw wierszom i kolumnom
colnames(x)<-
c("v1","v2","v3","v4","v5","v6","v7","v8","v9")
rownames(x)<-nazwy
# Zapisanie danych do pliku
write.table(x, file="01_dane_przestrzenne.csv",
sep=";", dec=",", row.names=TRUE, col.names=NA)
```

4. Procedury automatycznego pozyskiwania danych – szeregi czasowe

Zmienne wymagane w badaniu (dane dla Polski z lat 2005-2017):
v1 – produkt krajowy brutto w mln zł (ceny bieżące),
v2 – nakłady inwestycyjne w mln zł (ceny bieżące),
v3 – pracujący w tys. osób,
v4 – produkcja sprzedana przemysłu w mln zł (ceny bieżące),
v5 – stopa bezrobocia rejestrowanego w %.

Zmienne dostępne w BDL oraz wymagane przeliczenia

Zmienna dostępne w BDL	ID	Wymagane przeliczenia
x1 – produkt krajowy brutto w mln zł (ceny bieżące)	458271	v1=x1 (nie wymaga)
x2 – nakłady inwestycyjne w tys. zł (ceny bieżące)	6551	v2=x2 (nie wymaga)
x3 – pracujący w tys. osób	479270	v3=x3 (nie wymaga)
x4 – produkcja sprzedana przemysłu w mln zł (ceny bieżące)	148632	v4=x4 (nie wymaga)
x5 – stopa bezrobocia rejestrowanego w %	60270	v5=x5 (nie wymaga)

4. Procedury automatycznego pozyskiwania danych – szeregi czasowe

```
library(bdl); library(tidyr); library(magrittr); library(dplyr)
# Indywidualny klucz BDL
options(bdl.api_private_key="??????????????????????????????")
# stałe
rok<-c(2005:2017)
no_uL=0 # (0 - Polska, 1 - makroregion, 2 - województwo, 3 - region, 4 - podregion, 5 - powiat, 6 - gmina, 7 - miejscowość)
# Wczytanie informacji o zmiennych z BDL
d1<-get_data_by_variable("458271",unitParentId=NULL, unitLevel=no_uL,year=rok)
d2<-get_data_by_variable("6551",unitParentId=NULL,unitLevel=no_uL,year=rok)
d3<-get_data_by_variable("479270",unitParentId=NULL,unitLevel=no_uL,year=rok)
d4<-get_data_by_variable("148632",unitParentId=NULL,unitLevel=no_uL,year=rok)
d5<-get_data_by_variable("60270",unitParentId=NULL,unitLevel=no_uL,year=rok)
# Połączenie zmiennnych w jedną ramkę danych
data <- d1 %>%
  full_join (d2, by=c("id","year")) %>%
  full_join (d3, by=c("id","year")) %>%
  full_join (d4, by=c("id","year")) %>%
  full_join (d5, by=c("id","year"))
x<-as.data.frame(data)
# Nadanie nazw wierszom i kolumnom
nazwy<-x[,3]
rownames(x)<-nazwy
x<-x[,c(4,7,10,13,16)]
colnames(x)<-c("v1","v2","v3","v4","v5")
# Zapisanie danych do pliku
write.table(x, file="02_szeregi_czasowe.csv", sep=";", dec=",", row.names=TRUE, col.names=NA)
```

4. Procedury automatycznego pozyskiwania danych – szeregi czasowo-przekrojowe (panelowe)

Zmienne wymagane w badaniu (dane według województw Polski dla lat 2005-2017):
v1 – produkt krajowy brutto w mln zł (ceny bieżące),
v2 – nakłady inwestycyjne w mln zł (ceny bieżące),
v3 – pracujący w tys. osób,
v4 – produkcja sprzedana przemysłu w mln zł (ceny bieżące),
v5 – stopa bezrobocia rejestrowanego w %.

Zmienne dostępne w BDL oraz wymagane przeliczenia

Zmienna dostępne w BDL	ID	Wymagane przeliczenia
x1 – produkt krajowy brutto w mln zł (ceny bieżące)	458271	v1=x1 (nie wymaga)
x2 – nakłady inwestycyjne w tys. zł (ceny bieżące)	6551	v2=x2 (nie wymaga)
x3 – pracujący w tys. osób	479270	v3=x3 (nie wymaga)
x4 – produkcja sprzedana przemysłu w mln zł (ceny bieżące)	148632	v4=x4 (nie wymaga)
x5 – stopa bezrobocia rejestrowanego w %	60270	v5=x5 (nie wymaga)

4. Procedury automatycznego pozyskiwania danych – szeregi czasowo-przekrojowe (panelowe)

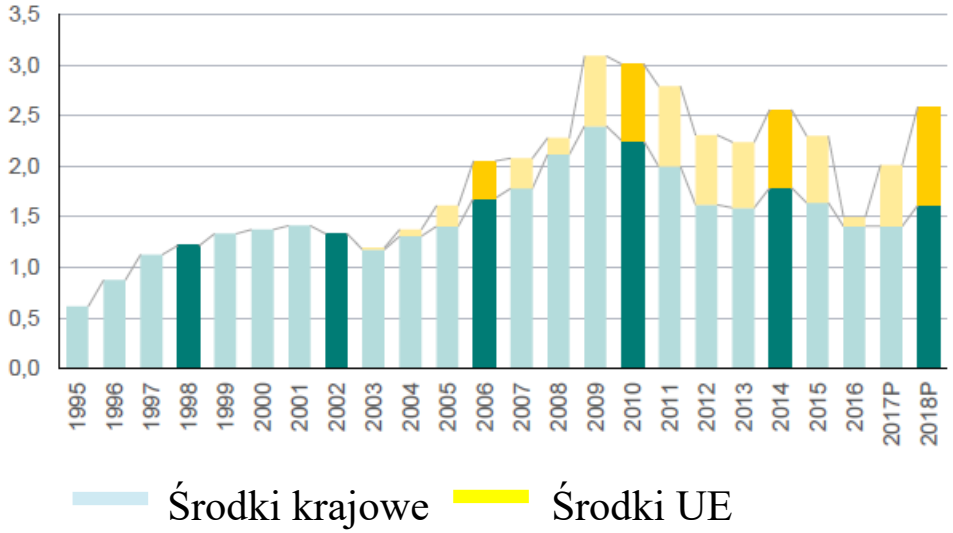
```
library(bdl); library(tidyr); library(magrittr); library(dplyr)
# Indywidualny klucz BDL
options(bdl.api_private_key="????????????????????????????????")
# stałe
rok<-c(2005:2017)
no_uL=2 # (0 - Polska, 1 - makroregion, 2 - województwo, 3 - region, 4 - podregion, 5 - powiat, 6 - gmina, 7 - miejscowość)
# Wczytanie informacji o zmiennych z BDL
d1<-get_data_by_variable("458271",unitParentId=NULL, unitLevel=no_uL,year=rok)
d2<-get_data_by_variable("6551",unitParentId=NULL,unitLevel=no_uL,year=rok)
d3<-get_data_by_variable("479270",unitParentId=NULL,unitLevel=no_uL,year=rok)
d4<-get_data_by_variable("148632",unitParentId=NULL,unitLevel=no_uL,year=rok)
d5<-get_data_by_variable("60270",unitParentId=NULL,unitLevel=no_uL,year=rok)
# Stworzenie struktury z kombinacjami lat i województw
id=get_units(level=2)$id
name=get_units(level=2)$name
year<-as.character(rep(rok,length(id)))
id<-rep(id,each=length(rok))
name<-rep(name,each=length(rok))
iny<-as.tbl(data.frame(id,name,year))
# Połączenie stworzonej struktury z pobranymi danymi (wraz z brakującymi danymi NA)
d1<-iny%>%left_join(d1)
d2<-iny%>%left_join(d2)
d3<-iny%>%left_join(d3)
d4<-iny%>%left_join(d4)
d5<-iny%>%left_join(d5)
```

4. Procedury automatycznego pozyskiwania danych – szeregi czasowo-przekrojowe (panelowe)

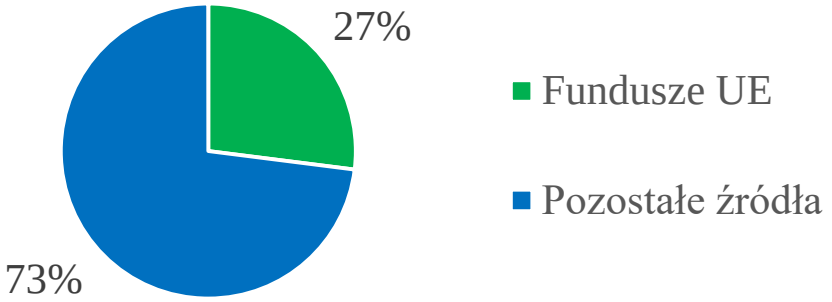
```
# Połączenie zmiennnych w jedną ramkę danych
data <- d1 %>%
  full_join (d2, by=c("id","year")) %>%
  full_join (d3, by=c("id","year")) %>%
  full_join (d4, by=c("id","year")) %>%
  full_join (d5, by=c("id","year"))
x<-as.data.frame(data)
# Nadanie nazw wierszom i kolumnom
x<-x[,c(2,3,4,7,10,13,16)]
colnames(x)<-c("województwo","rok","v1","v2","v3","v4","v5")
print(as.tbl(x),n=1000)
# Zapisanie danych do pliku
write.table(x, file="03_szeregi_czasowo-przestrzenne_v1.csv", sep=";", dec=",", row.names=TRUE, col.names=NA)
```

5. Zastosowanie z wykorzystaniem drzew regresyjnych

Wydatki inwestycyjne JST są mocno zależne od cyklu politycznego



Od 2004 roku są współfinansowane przez środki Unii Europejskiej

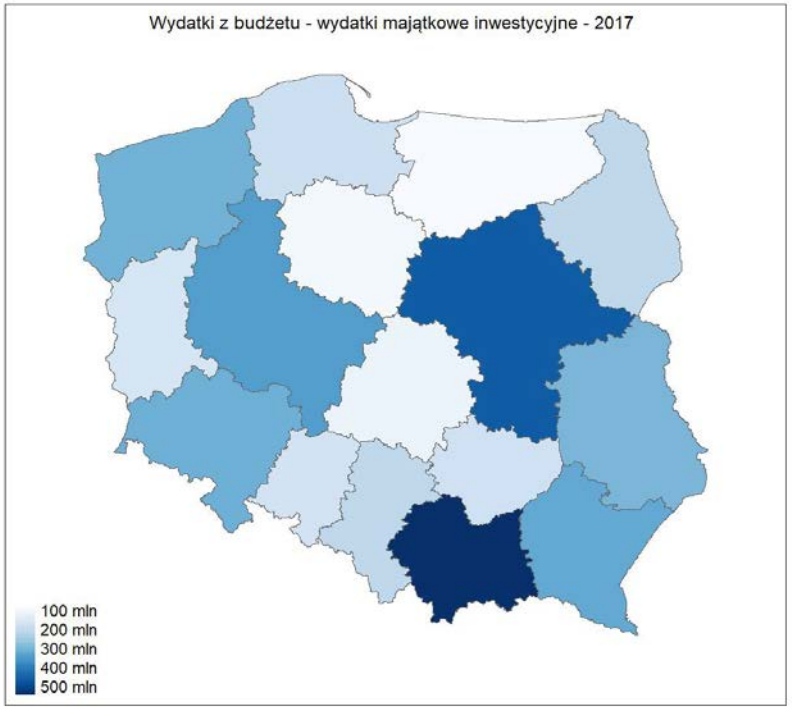


5. Zastosowanie z wykorzystaniem drzew regresyjnych

Rozkład wydatków inwestycyjnych województw jest nierównomierny

Wykres przygotowano za pomocą pakietu bdl.maps

```
library(bdl.maps)
generate_map(varId="6478", unitLevel=2,
      year="2017")
```



Do pozyskania danych wykorzystano opisaną wcześniej procedurę dla danych czasowo-przestrzennych i zaimportowano dane:

Y – wydatki majątkowe województw w mln zł
X1 – dochody własne w mln zł
X2 – dotacje z budżetu państwa w mln zł
X3 – środki z UE w mln zł
X4 – zadłużenie w mln zł
X5 – obsługa długu w mln zł
X6 – limit_1 (X4/X1<0,6)
X7 – limit_2 (X5/X1<0,15)

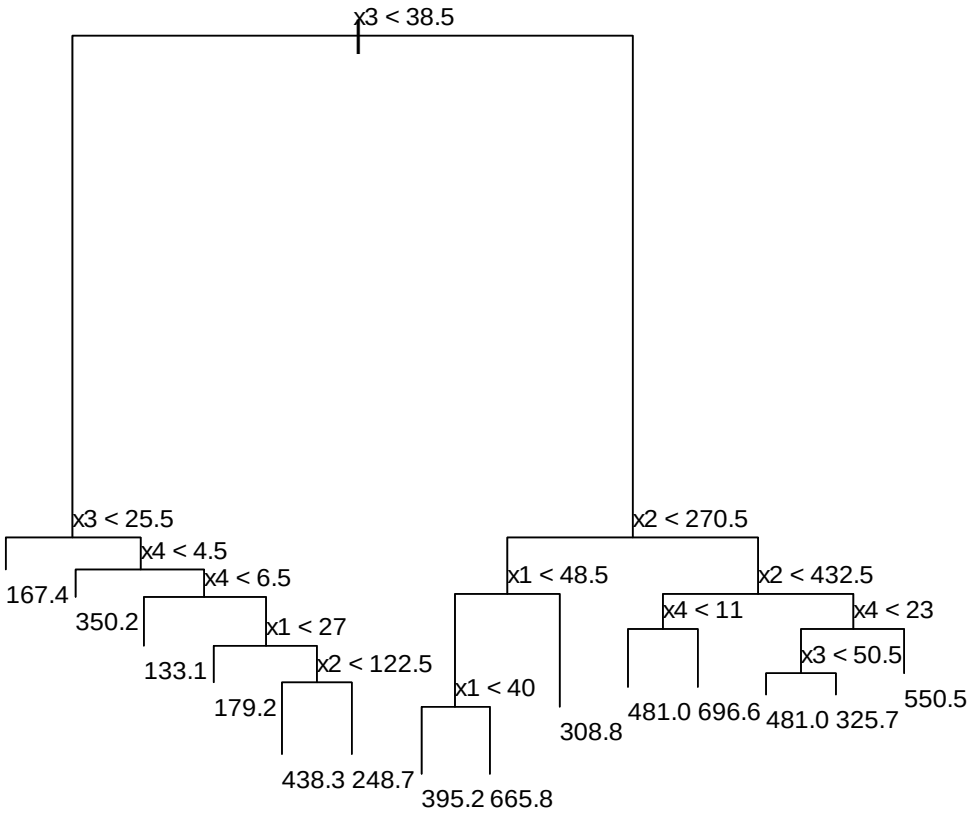
Lata 2004-2017 były okresem, kiedy obowiązywały limity zadłużenia:
Dług < 60% dochodów oraz
Obsługa długu < 15% dochodów

5. Zastosowanie z wykorzystaniem drzew regresyjnych

Zbudowano model dla zmiennej Y w postaci drzewa regresyjnego ze względu na zmienne X1-X7 (dane dla 16 województw z lat 2004-2017)

```
library(tree)
model <- tree(Y~., data=dane_woj)
print(model)
plot(model)
text(model, cex=0.7)
model.pr<-prune.tree(model,best=3)
print(model.pr)
```

```
1) root 126 4871000 364.1
  2) x3 < 38.5 64 1184000 257.0 *
  3) x3 > 38.5 62 2193000 474.7
    6) x2 < 270.5 25 1049000 411.8
     12) x1 < 48.5 11  466400 542.8 *
     13) x1 > 48.5 14  245000 308.8 *
    7) x2 > 270.5 37  978700 517.2 *
```



Zmienna o największym wpływie na poziom inwestowania przez województwa to środki z Unii Europejskiej.

6. Podsumowanie

W referacie zaprezentowano:

1. Nowy sposób automatycznego pozyskiwania danych z Banku Danych Lokalnych z wykorzystaniem pakietu bdl oraz interfejsu API (Application Programming Interface)
2. Architekturę interakcji BDL<->API<->pakiet bdl <-> program R
3. Przykłady i skrypty programu R pozwalające na automatyczne pozyskiwanie danych dla szeregów przekrojowych, czasowych, oraz czasowo-przekrojowych
4. Przykład z wykorzystaniem drzewa regresyjnego dla wydatków inwestycyjnych województw Y w latach 2004-2017 w zależności od 7 zmiennych

Dziękujemy za uwagę