KAPITOLA 2

Práce s daty

V této kapitole:
- Datové zdroje
- Čištění dat
- Datové formáty
- Začínáme s OpenRefine

Abyste mohli data skutečně analyzovat, budete potřebovat, aby byla přesná. V této kapitole si řekneme, jak získat, vyčistit, normalizovat a převést surová data do standardního formátu, jakým je například CSV či JSON, pomocí OpenRefine.

V této kapitole se budeme věnovat následujícím tématům:

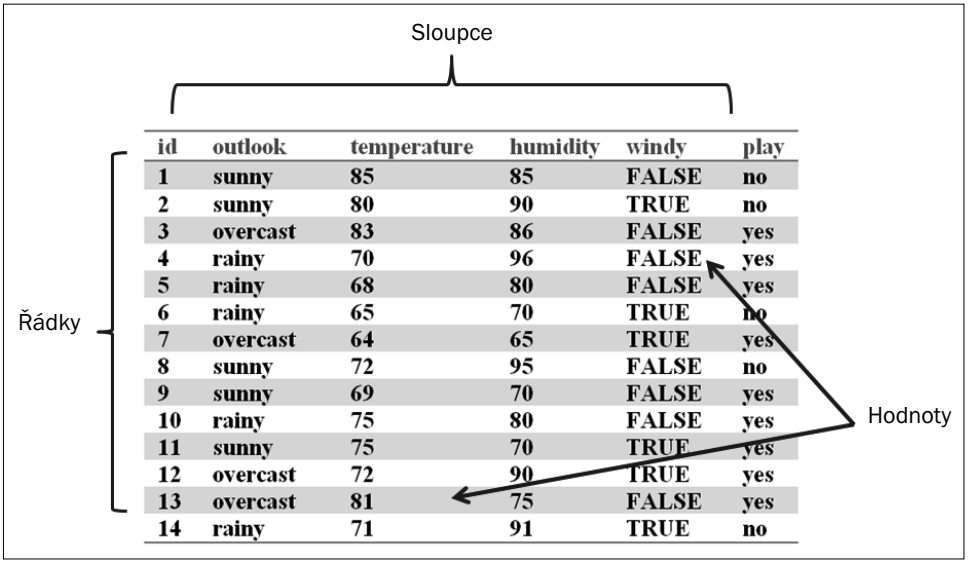
- Datové zdroje
  - Otevřená data
  - Textové soubory
  - Soubory aplikace Excel
  - SQL databáze
  - NoSQL databáze
  - Multimédia
  - Data z webu
- Čištění dat
  - Statistické metody
  - Rozložení textu
  - Převádění dat
- Datové formáty
  - CSV
  - JSON
  - XML
  - YAML
- Začínáme s OpenRefine

Datové zdroje

Datový zdroj je pojem, kterým se označují veškeré technologie, jež se pojí se získáváním a ukládáním dat. Datovým zdrojem může být cokoliv od jednoduchého textového souboru až po velkou databázi. Surová data mohou pocházet ze sledovacích protokolů, senzorů, transakcí a z uživatelských akcí.

V tomto oddíle se podíváme na nejběžnější formy datových zdrojů a sad.

Datová sada je sbírkou dat, která je obvykle zastoupena tabulkou. Každý sloupec takové tabulky představuje konkrétní proměnnou a každý řádek odpovídá jednomu dílku dat, viz následující obrázek:



id	outlook	temperature	humidity	windy	play
1	sunny	85	85	FALSE	no
2	sunny	80	90	TRUE	no
3	overcast	83	86	FALSE	yes
4	rainy	70	96	FALSE	yes
5	rainy	68	80	FALSE	yes
6	rainy	65	70	TRUE	no
7	overcast	64	65	TRUE	yes
8	sunny	72	95	FALSE	no
9	sunny	69	70	FALSE	yes
10	rainy	75	80	FALSE	yes
11	sunny	75	70	TRUE	yes
12	overcast	72	90	TRUE	yes
13	overcast	81	75	FALSE	yes
14	rainy	71	91	TRUE	no

Datová sada představuje fyzickou implementaci datového zdroje. Nejčastěji nese datová sada následující znaky:

- Charakteristiky datové sady (má dvě nebo více proměnných)
- Počet instancí
- Oblast (například život, obchod apod.)
- Znaky atributů (data jsou skutečná, kategorická nebo číselná)
- Počet atributů
- Přiřazené úkony (například klasifikace a rozdělení do clusterů)
- Chybějící hodnoty

Otevřená data

Otevřená data představují data, která lze opakovaně používat a distribuovat bez omezení. Níže uvádím krátký seznam úložišť a databází otevřených dat:

- Datahub najdete na adrese *http://datahub.io/*
- Datovou sadu pro směnu knih (Book-Crossing) najdete na *http://www.informatik.unifreiburg.de/~cziegler/BX/*
- Světovou zdravotnickou organizaci najdete na adrese *http://www.who.int/research/en/*
- Světovou banku najdete na adrese *http://data.worldbank.org/*
- Agenturu NASA najdete na adrese *http://data.nasa.gov/*
- Vládu Spojených států amerických najdete na adrese *http://www.data.gov/*
- Datové sady strojového učení najdete na adrese *http://bitly.com/bundles/bigmlcom/2*
- Vědecká data z münsterské univerzity najdete na adrese *http://data.uni-muenster.de/*
- Kvalitativní datové sady výzkumu Hilary Masonové najdete na adrese *https://bitly.com/bundles/hmason/1*

Tip: Další zajímavé zdroje dat pochází ze soutěží v těžbě dat a poznávání, například ACM-KDD Cup a Kaggle. Ve většině případů jsou datové sady dostupné i poté, co soutěž skončí.

Data ze soutěže ACM-KDD Cup najdete na adrese *http://www.sigkdd.ort/kddcup/index.php*.

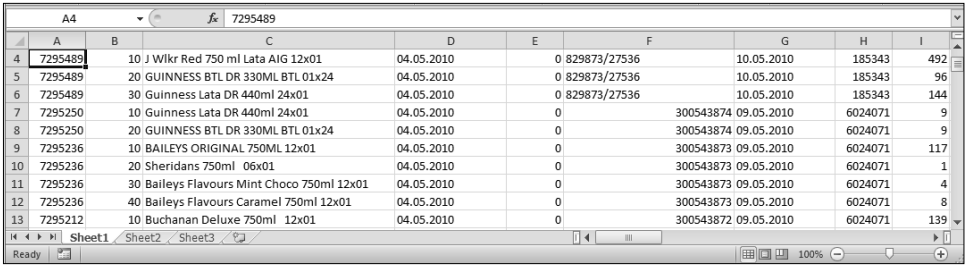
Data ze soutěže Kaggle najdete na *http://www.kagle.com/competitions*.

Textové soubory

Data se běžně ukládají do textových souborů, protože ty se snadno převádí do jiných formátů a často se s nimi lépe obnovuje zpracování než s jinými formáty. Velká množství dat pochází z protokolů, senzorů, e-mailů a transakcí. Textové soubory najdeme v několika formátech, například CSV (data oddělená čárkami), TXV (data oddělená tabulátorem), **XML** (Extensible Markup Language) a JSON (viz oddíl Datové formáty).

Soubory aplikace Excel

Pravděpodobně nejužívanějším a nejvíce podceňovaným nástrojem pro datovou analýzu je Microsoft Excel. Pravdou je, že Excel nabízí několik velmi dobrých funkcí, například filtrování, agregační funkce a pomocí aplikace Visual Basis for Application umí také vytvářet **SQL** – například dotazy na tabulky a externí databáze.



	A	B	C	D	E	F	G	H	I
4	7295489	10	J Wlkr Red 750 ml Lata AIG 12x01	04.05.2010	0	829873/27536	10.05.2010	185343	492
5	7295489	20	GUINNESS BTL DR 330ML BTL 01x24	04.05.2010	0	829873/27536	10.05.2010	185343	96
6	7295489	30	Guinness Lata DR 440ml 24x01	04.05.2010	0	829873/27536	10.05.2010	185343	144
7	7295250	10	Guinness Lata DR 440ml 24x01	04.05.2010	0	300543874	09.05.2010	6024071	9
8	7295250	20	GUINNESS BTL DR 330ML BTL 01x24	04.05.2010	0	300543874	09.05.2010	6024071	9
9	7295236	10	BAILEYS ORIGINAL 750ML 12x01	04.05.2010	0	300543873	09.05.2010	6024071	117
10	7295236	20	Sheridans 750ml 06x01	04.05.2010	0	300543873	09.05.2010	6024071	1
11	7295236	30	Baileys Flavours Mint Choco 750ml 12x01	04.05.2010	0	300543873	09.05.2010	6024071	4
12	7295236	40	Baileys Flavours Caramel 750ml 12x01	04.05.2010	0	300543873	09.05.2010	6024071	8
13	7295212	10	Buchanan Deluxe 750ml 12x01	04.05.2010	0	300543872	09.05.2010	6024071	139

Excel nám nabízí několik vizualizačních nástrojů. Možnosti analýzy (ve verzi 2010) můžeme rozšířit instalací nástroje Analysis ToolPak, který umožňuje provádět regresi, korelaci, kovarianci, Fourierovu analýzu apod. Další informace o nástroji Analysis ToolPak najdete na adrese *http://bit.ly/ZQKwSa*.

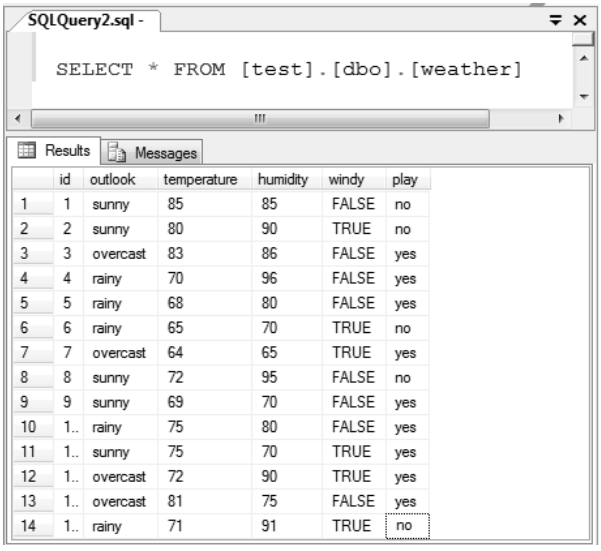
Mezi nevýhody Excelu patří fakt, že nedokáže konzistentně zacházet s chybějícími hodnotami a také že nezaznamenává údaje o způsobu provedení analýzy. S nástrojem Analysis ToolPak budete muset zpracovávat jednotlivé tabulky zvlášť. Proto je jinde než u jednoduchých příkladů z hlediska statistické analýzy jen chabým nástrojem.

Soubory Excelu (.xls) můžeme snadno převést do jiného textového formátu, například CSV, TSV, či dokonce XML. Tabulku vyexportujete v aplikaci Excel tak, že klepnete na nabídku **Soubor** (File), vyberete možnost **Uložit a odeslat** (Save & Send) a v sekci **Změnit typ souboru** (Change File Type) vyberete upřednostňovaný formát, například **CSV** (data oddělená čárkou).

SQL databáze

Databáze je organizovanou sbírkou dat. SQL je databázový jazyk určený ke správě a manipulaci dat v systémech **RDBMS** (Relational Database Management Systems). Systém **DBMS** (Database Management Systems) zodpovídá za správu integrity a zabezpečení uložených dat, ale také za obnovu informací v případech, kdy systém selže. Jazyk SQL obsahuje dvě podmnožiny příkazů: **DLL** (Data Definition Language) a **DML** (Data Manipulation Language).

Data jsou řazena do schémat (databáze) a rozdělena do tabulek na základě logických vazeb. Načítat je můžeme z databáze pomocí dotazů na hlavní schéma, viz následující snímek:



	id	outlook	temperature	humidity	windy	play
1	1	sunny	85	85	FALSE	no
2	2	sunny	80	90	TRUE	no
3	3	overcast	83	86	FALSE	yes
4	4	rainy	70	96	FALSE	yes
5	5	rainy	68	80	FALSE	yes
6	6	rainy	65	70	TRUE	no
7	7	overcast	64	65	TRUE	yes
8	8	sunny	72	95	FALSE	no
9	9	sunny	69	70	FALSE	yes
10	1..	rainy	75	80	FALSE	yes
11	1..	sunny	75	70	TRUE	yes
12	1..	overcast	72	90	TRUE	yes
13	1..	overcast	81	75	FALSE	yes
14	1..	rainy	71	91	TRUE	no

Jazyk DDL nám umožňuje vytvářet, mazat a upravovat tabulky databáze. Rovněž si prostřednictvím něho můžeme definovat klíče, které definují vztahy mezi tabulkami a implementují omezení mezi jednotlivými tabulkami.

- `CREATE TABLE`: Tento příkaz vytvoří novou tabulku.
- `ALTER TABLE`: Tento příkaz upraví tabulku.
- `DROP TABLE`: Tento příkaz odstraní tabulku.

Jazyk DML umožní uživatelům přistupovat k datům a nakládat s nimi.

- `SELECT`: Tento příkaz načte data z databáze.
- `INSERT INTO`: Tento příkaz vloží do databáze nová data.
- `UPDATE`: Tento příkaz upraví data v databázi.
- `DELETE`: Tento příkaz odstraní data z databáze.

NoSQL databáze

Pojem NoSQL (Not only SQL, nejen SQL) využívá několik technologií, v nichž povaha dat nevyžaduje relační model. Technologie NoSQL umožňuje pracovat s ohromnými datovými objemy zajišťuje vyšší stabilitu, škálovatelnost a výkon.

Rozšířené příklady databáze pro uchovávání dokumentů (MongoDB) najdete v kapitole 12, Zpracování a agregace dat v MongoDB, a v kapitole 13, Práce s modelem MapReduce.

Nejběžnější typy datových úložišť NoSQL jsou:

- **Úložiště dokumentů** – Data se uchovávají a třídí jako sbírky dokumentů. Schéma modelu je flexibilní. Všechny sbírky dokáží obsloužit libovolný počet polí. Například databáze MongoDB pracuje s typem BSON (binární JSON) a CouchDB používá dokumenty ve formátu JSON.
- **Úložiště klíčů a hodnot** – Data se uchovávají jako dvojice klíčů a hodnot bez předdefinovaného schématu. Hodnoty se načítají z klíčů. Příkladem jsou Apache Cassandra, Dynamo, Hbase a Amazon SimpleDB.
- **Úložiště založené na grafech** – Data se uchovávají v grafových strukturách s uzly, hranicemi a vlastnostmi na základě teorie grafů pro ukládání a načítání dat. Tyto typy databází dokáží skvěle zachytit vztahy v sociální síti. Příkladem jsou Neo4js, InfoGrid a Horton.

Další informace o databázích NoSQL najdete na adrese *http://nosql-database.org/*.

Multimédia

Díky narůstajícímu počtu mobilních zařízení se prioritou datové analýzy stává schopnost extrahovat sémantické informace z multimediálních datových zdrojů. Mezi datové zdroje řadíme přímo sledovatelná média, jakými jsou například hudba, obraz a video. Mezi aplikace tohoto typu datových zdrojů patří:

- Načítání obrázků na základě obsahu
- Načítání videa na základě obsahu
- Třídění filmů a videa
- Rozpoznání tváří
- Rozpoznání řeči
- Třídění zvuku a hudby

V kapitole 5, Rozpoznávání podobných obrázků, si ukážeme stroj vyhledávající podobné obrázky, který využívá databázi Caltech256, což je datová sada s více než 30 600 obrázky.

Data z webu

Když potřebujete najít nějaká data, bývá dobrým startovním bodem web. Při získávání dat z webu zpracováváme HTML kód webové stránky a z něj načítáme data, s nimiž pracujeme. Aplikace, které toto provádí, simulují osobu, jež si webovou stránku prohlíží v prohlížeči. V následujícím příkladu budeme předpokládat, že chceme zjistit aktuální cenu zlata ze stránky *www.gold.org*, viz následující snímek obrazovky:

Následně budeme muset na stránce prozkoumat prvek **Gold Spot Price**, přičemž najdeme následující HTML příznak:

```
<td class="value" id="spotpriceCellAsk">1,573.85</td>
```

V příznaku `td` si všimneme identifikátoru `id`, `spotpriceCellAsk`. Tento element načteme pomocí následujícího kódu v Pythonu.

Tip: V tomto příkladě použijeme 4. verzi knihovny BeautifulSoup. Na Linuxu si ji můžete nainstalovat ve správci balíčkovacího systému. Spusťte terminál a zadejte následující příkaz:

```
$ apt-get install python-bs4
```

Na Windows si musíte stáhnout knihovnu z následujícího odkazu *http://crummy.com/software/BeautifulSoup/bs4/download/*.

Instalaci provedete zadáním příkazu:

```
$ python setup.py install
```

1. Nejprve musíte naimportovat knihovny `BeautifulSoup` a `urllib.request`.

```
from bs4 import BeautifulSoup
import urllib.request
from time import sleep
from datetime import datetime
```

2. Následně použijete funkci `getGoldPrice`, kterou z webové stránky načtete aktuální cenu. K tomu budete muset zadat adresu URL, jež zadá dotaz a načte celou stránku.

```
Req = urllib.request.urlopen(url)
page = req.read()
```

3. Následně pomocí knihovny `BeautifulSoup` rozložíte stránku (vytvoříte seznam všech elementů na stránce) a vyžádáte si element `td` s `id spotpriceCellAsk`:

```
scraping = BeautifulSoup(page)
price= scraping.findAll("td",attrs={"id":"spotpriceCellAsk"})[0].
text
```

4. Nyní vrátíme proměnnou `price` s aktuální cenou zlata. Tato hodnota se bude na stránce každou minutu měnit. My budeme chtít všechny hodnoty za celou hodinu, takže funkci

`getGoldPrice` zavoláme ve smyčce 60x a po každém volání necháme skript 59 sekund čekat.

```
for x in range(0,60):
...
  sleep(59)
```

5. Nakonec uložíme výsledek do výstupního souboru `goldPrice.out` a přidáme aktuální čas ve formátu HH:MM:SS (dopoledne – AM nebo odpoledne – PM), například 11:35:42PM, oddělený čárkou.

```
with open("goldPrice.out","w") as f:
...
        sNow = datetime.now().strftime("%I:%M:%S%p")
        f.write("{0}, {1} \n ".format(sNow, getGoldPrice()))
```

Funkce `datetime.now().strftime` vytvoří řetězec představující čas pod vedením explicitního formátovacího řetězce `"%I:%M:%S%p"`, ve kterém `%I` představuje hodinu coby desítkové číslo od 0 do 12, `%M` zastupuje minuty coby desítkové číslo od 00 do 59, `%S` představuje sekundy coby desítkové číslo od 00 do 61 a `%p` představuje buď dopoledne (A.M.), nebo odpoledne (P.M.).

Kompletní seznam formátovacích pravidel najdete na adrese *http://docs.python.org/3.2/library/datetime.html*.

Zde vidíte kompletní skript:

```
from bs4 import BeautifulSoup
import urllib.request
from time import sleep
from datetime import datetime
def getGoldPrice():
    url = "http://gold.org"
    req = urllib.request.urlopen(url)
    page = req.read()
    scraping = BeautifulSoup(page)
    price= scraping.findAll("td",attrs={"id":"spotpriceCellAsk"})[0]
    .text
    return price

with open("goldPrice.out","w") as f:
    for x in range(0,60):
        sNow = datetime.now().strftime("%I:%M:%S%p")
        f.write("{0}, {1} \n ".format(sNow, getGoldPrice()))
        sleep(59)
```

Tip: Kompletní skript (WebScraping.py) si můžete stáhnout z autorova úložiště na GitHubu, a to na adrese *http:/github.com/hmcuesta/PDA_Book/tree/master/Chapter2*.

Výstupní soubor, `goldPrice.out`, bude vypadat následovně:

```
11:35:02AM, 1481.25
11:36:03AM, 1481.26
11:37:02AM, 1481.28
```

```
11:38:04AM, 1481.25
11:39:03AM, 1481.22
...
```

Čištění dat

Čištění dat je proces opravy a odstraňování chybných, neúplných, nesprávně formátovaných a duplicitních dat z datové sady.

Výsledek datové analýzy nezávisí jen na algoritmech, ale také na kvalitě dat. A právě proto bude dalším krokem získávání dat jejich čištění. Abychom se vyhnuli chybným datům, měla by datová sada nést následující charakteristiky:

- Správnost
- Úplnost
- Přesnost
- Konzistence
- Jednotnost

Špatná data najdete pomocí některého z jednoduchých validačních statistických postupů, ale také rozložením textů a odstraněním duplicitních hodnot. Chybějící a neúplná data vás mohou dovést k vysoce zavádějícím výsledkům.

Statistické metody

Tento postup si žádá alespoň částečnou znalost kontextu. Bez něj bychom nedokázali najít neočekávaná, a tedy chybná data, a to ani v případě, že by odpovídala datovým typem, ale hodnotou byla mimo rozsah. Řešením by bylo nastavit hodnoty na průměrné nebo středové hodnoty. Pomocí statistické validace můžeme ošetřit chybějící hodnoty, které můžeme buď nahradit jednou z pravděpodobných hodnot zjištěnou interpolací, nebo zredukovat datovou sadu pomocí decimování.

- **Aritmetický průměr** – Tato hodnota se vypočítá sečtením všech hodnot a podělením výsledku počtem hodnot.
- **Medián** – Medián je střední hodnota řazeného seznamu hodnot.
- **Rozmezí** – Čísla a data by měla spadat do určitého rozhraní. To znamená, že mají danou minimální a maximální hodnotu.
- **Clustering** – Data získaná přímo od uživatele bývají zpravidla mnohoznačná anebo kvůli chybě odkazují na stejnou hodnotu. Například hodnoty „Buchanan Deluxe 750 ml 12 x 01“ a „Buchanan Deluxe 750 ml 12 x 01.“ se liší pouze tečkou. Případně hodnoty Microsoft, MS a Microsoft Corporation odkazují na stejnou společnost a všechny jsou platné.