

Open Source Workshop

Data analyse met (Python) Pandas

DeNederlandscheBank

EUROSYSTEEM

Data analysis met Pandas

- 14:00 Theorie/achtergrond informatie
- 14:15 Gebruik Pandas
- 15:15 Voorbereiden case study
- 15:30 Case study uitvoeren
- 16:45 Resultaten presenteren en bespreken



Wat is Pandas?

"Pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language."

Een Python package, dat gebouwd is boven op het NumPy package

- Beperkt het schrijven van code Python voor het ontsluiten, manipuleren en verwerken van gestructureerde data
- Makkelijke visualisatie via Matplotlib
- Slechts 2 data structuren: Series en DataFrame



Pandas DataFrame

DNBUNRESTRICTED

Tabel met 1 of twee dimensies:

- Series: 1-dimensionaal
- DataFrames: 2-dimensionaal (axis=0 zijn de rijen, axis=1 zijn de kolommen)
 - DataFrame is opgebouwd als meerdere series met 1 index

Gebruik van makkelijk te interpreteren namen voor de kolommen/rijen

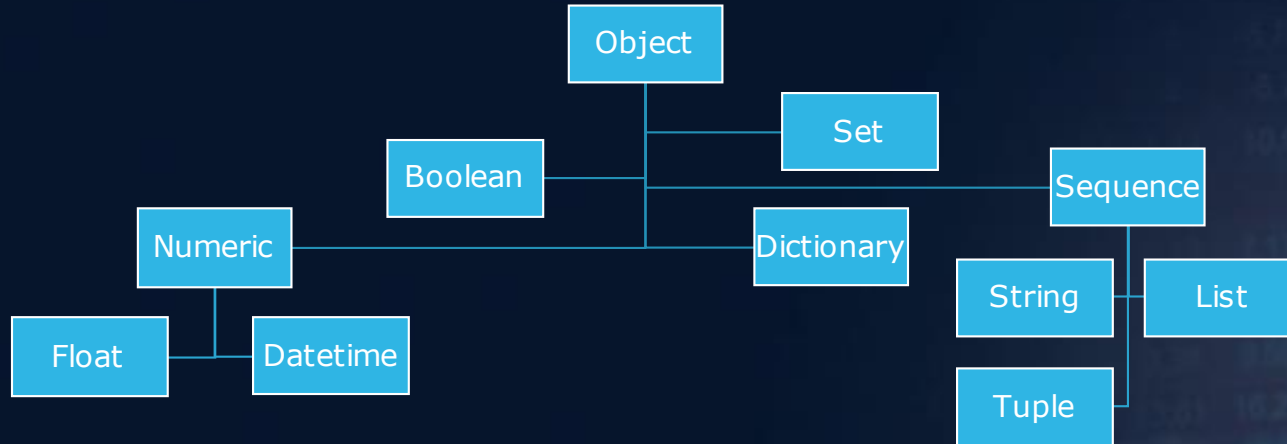
- Kolommen \simeq Variabelen
- Rijen \simeq Observaties

Overige functionaliteit:

- Lege data velden worden NaN (np.nan).
- Toevoegen/verwijderen van kolommen in een DataFrame moet makkelijk gaan
- Data samenvoegen gebaseerd op de indexes
- Zijn niet gemaakt om van vorm te veranderen

Datatypes in Python

DNBUNRESTRICTED



Data combineren

DNBUNRESTRICTED

A

B

Tabellen A en B willen we samenvoegen op basis van
gelijke waarden uit specifieke kolommen.



Outer

Alle datapunten meenemen



Left/right

Alle datapunten uit de
linker/rechter helft meenemen
en de overlappende data



Inner

Alleen de overlappende data



Als de matches niet uniek zijn zullen alle combinaties terug komen in het
samengevoegde dataframe

Data analysis met Pandas: Toepassing

- Verschillende soorten data en databronnen

- Wat is een dataframe
- Data inlezen uit verschillende bronnen voordoen (`.read_csv`, `.read_excel`, `.read_json`)
- Inlezen van meerdere bestanden
- Data types (`.dtypes`, `.astype`, `.head`)

- Data schonen

- Ongeldige en ontbrekende waarden (`.isna`, `.isnull`, `.drop_duplicates`, `.drop_na`)
- Veel voorkomende datafouten (`.describe`, `.min/max`, `.quantile`, `.unique`)
- Data preprocessing (`.round`)

- Data combineren

- Indexing en filteren in dataframe (`.iloc`, `.loc`, `.at`, chain indexing)
- Dataframes met elkaar combineren (`.merge`, `.concat`, `regex`)

- Data analyseren

- Transformatie van data (`.sort`, `.drop`, `.add`, `.pivot_table`, `.melt`, `.stack`, `.unstack`, `.crosstab`)
- Dataframe berekeningen kolom/rij gebaseerd, itereren (`.eval`, `.apply`, `.iterrows`)
- Geaggregeerde statistieken (`.groupby`)
- Plotten van dataframe (`.plot`, `.plot.hist`, `.plot.bar`)

- Data wegschrijven

- Verschillende opties (`.to_csv`, `.to_excel`, `.to_pickle`)

Data analysis met Pandas: Case study

Covid maatregelen in VS:

- Paycheck Protection Program: Leningen aan kleinbedrijf om mensen in dienst te houden

Zijn deze leningen op de juiste plek terecht gekomen?

- In de sectoren waar het aantal uren dat er gewerkt is het meeste achteruit ging

Extra:

- Op de locaties waar de afname in mobiliteit het grootst was

Data analysis met Pandas: Case study

Vijf databestanden:

- PPP_Loan_Data.csv De individuele leningen uit het PPP (voor grote leningen is enkel een loan_range bekend)
- US_Hours_Worked.json Geaggregeerde aantal gewerkte uren per week per maand vanaf 2019 per sector in duizendtallen seizoensgecorrigeerd
- NAICS.json Sectornamen die bij de sectorcodes (NAICS) horen

Code:

- Workshop_code.ipynb Om de data in te lezen vanuit Google Colab
- OSW_Pandas_us_hours_worked_verwerkt.ipynb Code om de unemployment data na het inlezen te verwerken

Locatie: https://github.com/mnijhuis-dnb/open_source_workshop

Extra:

- Global_Mobility_Report.csv Dagelijkse Google mobility indexes vanaf februari
- uszipcodes.xlsx Lijst met postcodes en census codes voor de VS