Mittuniversitetet
MID SWEDEN UNIVERSITY

DT094G – Jobba med datastrukturer i Pandas

Lennart Franked

5 april 2023

Mittuniversitetet
MID SWEDEN UNIVERSITY

Innehållsförteckning

Användning inom Systemadministration och Nätverk

- Nödvändigt när man jobbar med automation
- Samla in och redovisa statisk
- Analysera nätverksdata

Datatyper

- Datatyp definierar vilken typ av data som lagras
- Integer (heltal)
- float (flyttal)
- Char (tecken)
- Str (sträng)

Dataformat

- En standard för hur man lagrar data
- Samling med regler kring syntax och semantik
- Underlättar möjligheten att strukturera upp sitt data på ett lättåtkomligt sätt.
- Möjliggör att det lättare går att utbyta data.
- JSON, XML, YAML

Datastrukturer

- Strukturerar, lagrar och hanterar en eller flera datatyper
- Stödjer att man utför komplexa operationer
- Primitiva datastrukturer
  - Integer
  - Float
  - Char
- Icke-primitiva datastrukturer
  - Lagrar samlingar av primitiva datastrukturer
  - Linjära datastrukturer
  - Icke-linjära datastrukturer

Linjära datastrukturer

- Datastruktur som lagrar data sekventiellt
- Exempel
  - Array
  - Länkad lista
  - Stack
  - Kö

Icke-linjära datastrukturer

- Ej linjärt
- Element kan kopplas samman på flera sätt
- Ingen tydligt ordning
- Exempel
  - Träd
  - Graf

Mittuniversitetet
MID SWEDEN UNIVERSITY

Innehållsförteckning

Pandas

Om Pandas

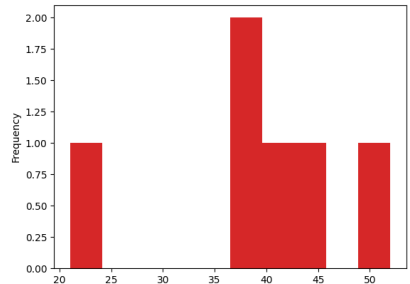
- Bibliotek i Python för att manipulera och analysera data [2].
- Statistik
- Liknande funktioner som man kan se i kalkylprogram, ex. Excel, Libre Calc...
- Användningsområden inkl
  - Läsa in och manipulera xls-filer
  - Läsa in och jobba med olika filformat, ex (xls, xml, latex, html, json, sql)
  - Visualisera data (grafer, tabeller)

Pandas Series

- Endimensionell datastruktur
- Stödjer etiketter på X-axeln
- Lagrar arrayer, listor, dict osv.
- Likt Python Dictionary, fast används inte enbart för att lagra.
- Se [1, Series] för komplett lista.
- Användningsområden varierar, men vanligt om man vill enkelt plocka ut statistik på en endimensionell dataserie.

Exempelkod

```
>>> import pandas as pd
>>> employee_ages = {'Måns': 38,
                     'Olliver':37,
                     'Eskil':44,
                     'Danny': 42,
                     'Anna': 52,
                     'Daria': 21
                    }
>>> employee_series = pd.Series(data=employee_ages)
>>> employee_series['Måns']
38
>>> employee_series['Anna']
52
>>> employee_series.mean()
39.0
>>> import matplotlib.pyplot as plt
>>> employee_series.plot(kind='hist')
<AxesSubplot:ylabel='Frequency'>
>>> plt.show()
```



Övning

- Skapa en Pandas Series från `'employee_ages'` från senast exemplet.
- Lägg till dig själv i slutet av denna.
- Med hjälp av Pandas Series inbyggda funktioner. Sortera listan baserat på ålder.
- Hur många år skiljer sig din ålder från medelåldern i listan?
- Använd Pandas dokumentation till din hjälp [1, Series].

employee_ages

```
1  employee_ages = {'Måns': 38,
2                   'Olliver':37,
3                   'Eskil':44,
4                   'Danny': 42,
5                   'Anna': 52,
6                   'Daria': 21
7                  }
```

Pandas Dataframe

- Tvådimensionell datastruktur
- Likt en tabell eller excel-kalkylark
- Pandas beskriver DataFrame som en Dict med Series-objekt [1, Dataframe].
- Lämplig att använda när man jobbar med kalkylark, eller data som strukturerats på liknande sätt.
- Enkelt att importera och exportera Dataframe till och från xlsx.

Läsa in Excel-ark

pandas.read_excel [1, read_excel]

```
pandas.read_excel(io, sheet_name=0, *, header=0, names=None, index_col=None, usecols=None,
  squeeze=None, dtype=None, engine=None, converters=None, true_values=None, false_values=None,
  skiprows=None, nrows=None, na_values=None, keep_default_na=True, na_filter=True, verbose=False,
  parse_dates=False, date_parser=None, thousands=None, decimal='.', comment=None, skipfooter=0,
  convert_float=None, mangle_dupe_cols=True, storage_options=None)
```

Intressanta argument

- **io:** str, bytes, ExcelFile, xlrd.Book, path object, or file-like object
- **sheet_name:** sheet_namestr, int, list, or None, default 0
- **header:** headerint, list of int, default 0

Kräver att xlrd eller openpyxl är installerat

Modifiera dataframes

Med hjälp utav Pandas dokumentation [1, Dataframe], ta reda på vad följande metoder gör.

Komma åt poster

- pandas.DataFrame.at
- pandas.DataFrame.iat
- pandas.DataFrame.loc
- pandas.DataFrame.iloc

Exempel läsa in Excel-ark

```
>>> from pandas import read_excel
>>> pandas_df = read_excel('names.xls', header=None)
>>> pandas_df.loc[0]
0                  Måns Hansson
1     Personnummer: 8510309159
2                      Rum S321
3                   Tel: 671420
Name: 0, dtype: object
>>> pandas_df.at[0,0]
'Måns Hansson'
>>> pandas_df.at[0,0] = 'Måns Persson'
>>> pandas_df.at[0,0]
'Måns Persson
```

Skriva dataframes till Excel

För att skriva till Excel används metoden:

pandas.ExcelWriter [1, ExcelWriter]

```
1 pandas.ExcelWriter(path, engine=None, date_format=None, datetime_format=None, mode='w',
2     storage_options=None, if_sheet_exists=None, engine_kwargs=None, **kwargs)
```

Intressanta argument

- **path:**
- **engine_kwargs**

Kräver att xlsxwriter eller openpyxl är installerat

Skriva dataframes till Excel

Konvertera dataframe till excel

pandas.DataFrame.to_excel [1, DataFrame_to_excel]

```
1 DataFrame.to_excel(excel_writer, sheet_name='Sheet1', na_rep='',
2     float_format=None, columns=None, header=True, index=True, index_label=None, startrow=0,
3     startcol=0, engine=None, merge_cells=True, encoding=_NoDefault.no_default, inf_rep='inf',
4     verbose=_NoDefault.no_default, freeze_panes=None, storage_options=None)
```

Intressanta argument

- **excel_writer**
- **header**
- **index**

Exempel skriva Excel-ark

```
1  >>> from pandas import ExcelWriter
2  >>> from pandas import read_excel
3  >>> pandas_df = read_excel('names.xls', header=None)
4  >>> pandas_df.at[0,0] = 'Måns␣Persson'
5  >>> with ExcelWriter('names_mod.xls') as writer:
6  ...     pandas_df.to_excel(writer, header=False, index=False)
7  ...
8  >>>
```

Referenser I

[1] *Pandas API reference*. 2023. URL: https://pandas.pydata.org/docs/reference/index.html.

[2] *Pandas User Guide*. 2023. URL: https://pandas.pydata.org/docs/user_guide/index.html.

Mittuniversitetet
MID SWEDEN UNIVERSITY