Kazan
Federal
University

HIGHER SCHOOL
for Information Technology
and Information Systems

Quick Introduction to the NumPy Library

NumPy Library (1)

```
import numpy as np
```

Creating NumPy Arrays, Loading and Saving Files

```
numpy_array = np.array(list)
```

```
array([[  7.4  ,   0.7  ,   0.   , ...,   0.56 ,   9.4  ,   5.   ],
       [  7.8  ,   0.88 ,   0.   , ...,   0.68 ,   9.8  ,   5.   ],
       [  7.8  ,   0.76 ,   0.04 , ...,   0.65 ,   9.8  ,   5.   ],
       ...,
       [  6.3  ,   0.51 ,   0.13 , ...,   0.75 ,  11.   ,   6.   ],
       [  5.9  ,   0.645,   0.12 , ...,   0.71 ,  10.2  ,   5.   ],
       [  6.   ,   0.31 ,   0.47 , ...,   0.66 ,  11.   ,   6.   ]])
```

NumPy Library (2)

Loading

```
numpy_array = np.genfromtxt("file.csv", delimiter=";",
skip_header=1)
```

Saving

```
np.savetxt('file.txt',arr,delimiter=' ')
```

```
np.savetxt('file.csv',arr,delimiter=',')
```

Generating random

```
np.random.rand(3,4)
```

```
np.random.rand(7,6)*100
```

NumPy Library (3)

Working and Inspecting Arrays

```
array.size
```

```
array.shape
```

```
array.dtype
```

```
array.tolist()
```

```
array.astype(dtype)
```

Indexing and Slicing

```
array[5]
```

```
array[0:5]
```

```
array[2,5]
```

```
array[0:5,4]
```

```
array[:2]
```

```
array[:,1]
```

Assignment

```
array[1]=4
```

```
array[:,10]=10
```

NumPy Library (4)

Sorting and Reshaping

```
array.sort()
```

```
array.sort(axis=0)
```

```
two_d_arr.flatten()
```

```
array.reshape(x,y)
```

```
array.resize((x,y))
```

Combining and Splitting

```
np.concatenate((array1,array2),axis=0)
```

```
np.concatenate((array1,array2),axis=1)
```

```
np.split(array,2)
```

```
np.hsplit(array,5)
```

NumPy Library (5)

Adding and Removing Elements

`np.append(array,values)` will append values to end of array.

`np.insert(array, 3, values)` will insert values into array before index 3

`np.delete(array, 4, axis=0)` will delete row on index 4 of array

`np.delete(array, 5, axis=1)` will delete column on index 5 of array

Descriptive Statistics

`np.mean(array,axis=0)` will return mean along specific axis (0 or 1)

`array.sum()` will return the sum of the array

`array.min()` will return the minimum value of the array

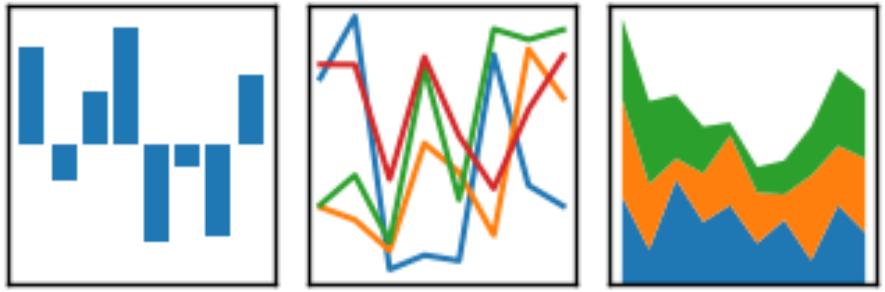
`array.max(axis=0)` will return the maximum value of specific axis

`np.var(array)` will return the variance of the array

`np.std(array,axis=1)` will return the standard deviation of specific axis

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



“Pandas” Python Data Analysis Library

Pandas Library (1)

Importing module

```
import pandas as pd
```

Loading and Saving Data with Pandas

- Convert a Python's list, dictionary or Numpy array to a Pandas data frame
- Open a local file using Pandas, usually a CSV file, but could also be a delimited text file (like TSV), Excel, etc
- Open a remote file or database like a CSV or a JSONon a website through a URL or read from a SQL table/database

```
pd.read_filetype()
```

Pandas Library (2)

Load data frame from file/connection `pd.read_filetype()`

Format Type	Data Description	Reader	Writer
text	CSV	read_csv	to_csv
text	JSON	read_json	to_json
text	HTML	read_html	to_html
text	Local clipboard	read_clipboard	to_clipboard
binary	MS Excel	read_excel	to_excel
binary	HDF5 Format	read_hdf	to_hdf
binary	Feather Format	read_feather	to_feather
binary	Msgpack	read_msgpack	to_msgpack
binary	Stata	read_stata	to_stata
binary	SAS	read_sas	
binary	Python Pickle Format	read_pickle	to_pickle
SQL	SQL	read_sql	to_sql
SQL	Google Big Query	read_gbq	to_gbq

Data frame from Python object `pd.DataFrame()`

Pandas Library (3)

```
df.to_filetype(filename)
```

Save data frame

DataFrame.from_csv(path[, header, sep, ...])	Read CSV file (DISCOURAGED, please use pandas.read_csv() instead).
DataFrame.from_dict(data[, orient, dtype])	Construct DataFrame from dict of array-like or dicts
DataFrame.from_items(items[, columns, orient])	Convert (key, value) pairs to DataFrame.
DataFrame.from_records(data[, index, ...])	Convert structured or record ndarray to DataFrame
DataFrame.info([verbose, buf, max_cols, ...])	Concise summary of a DataFrame.
DataFrame.to_pickle(path[, compression])	Pickle (serialize) object to input file path.
DataFrame.to_csv([path_or_buf, sep, na_rep, ...])	Write DataFrame to a comma-separated values (csv) file
DataFrame.to_hdf(path_or_buf, key, **kwargs)	Write the contained data to an HDF5 file using HDFStore.
DataFrame.to_sql(name, con[, flavor, ...])	Write records stored in a DataFrame to a SQL database.
DataFrame.to_dict([orient])	Convert DataFrame to dictionary.
DataFrame.to_excel(excel_writer[, ...])	Write DataFrame to an excel sheet
DataFrame.to_json([path_or_buf, orient, ...])	Convert the object to a JSON string.
DataFrame.to_html([buf, columns, col_space, ...])	Render a DataFrame as an HTML table.
DataFrame.to_feather(fname)	write out the binary feather-format for DataFrames
DataFrame.to_latex([buf, columns, ...])	Render an object to a tabular environment table.
DataFrame.to_stata(fname[, convert_dates, ...])	A class for writing Stata binary dta files from array-like objects
DataFrame.to_msgpack([path_or_buf, encoding])	msgpack (serialize) object to input file path
DataFrame.to_gbq(destination_table, project_id)	Write a DataFrame to a Google BigQuery table.
DataFrame.to_records([index, convert_datetime64])	Convert DataFrame to record array.
DataFrame.to_sparse([fill_value, kind])	Convert to SparseDataFrame
DataFrame.to_dense()	Return dense representation of NDFrame (as opposed to sparse)
DataFrame.to_string([buf, columns, ...])	Render a DataFrame to a console-friendly tabular output.
DataFrame.to_clipboard([excel, sep])	Attempt to write text representation of object to the system clipboard This can be pasted into Excel, for example.

Pandas Library (4)

Viewing and Inspecting Data

`df.mean()` Returns the mean of all columns

`df.corr()` Returns the correlation between columns in a data frame

`df.count()` Returns the number of non-null values in each data frame column

`df.max()` Returns the highest value in each column

`df.min()` Returns the lowest value in each column

`df.median()` Returns the median of each column

`df.std()` Returns the standard deviation of each column

Pandas Library (5)

Filter, Sort and Groupby

```
df[df[year] > 1984]
```

```
df.sort_values(col1)
```

```
df.sort_values(col2,ascending=False)
```

```
df.sort_values([col1,col2],ascending=[True,False])
```

Казанский федеральный
УНИВЕРСИТЕТ

Thanks For Your Attention!