

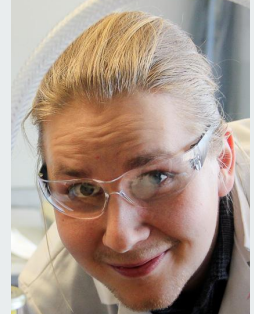


Databearbetning

Steget innan datavetenskap

Lektion 5 - Visualisering, WebScraping, Uppg 2 och 3

Dennis Biström
bistromd@arcada.fi



2 veckor kvar! - 3 lektioner, 3 uppg kvar



Upplägg

Föreläsningar med exempel - Var på plats, följ med!

Videoföreläsningar (60–120 min) att se på hemma

Veckouppgifter med deadline **varje** vecka.

Inget kodtilfälle! Använd F369 och fråga kaveri?

Kursverktyg

Python

Pandas (Python Data Analysis Library)

Jupyter Notebook

Installering: Anaconda (Linux / Mac / Windows)

<https://www.continuum.io/anaconda-overview>

Bedömning

Vitsordet bestäms på basis av era lösningar på kursuppgifterna. Maxpoäng 110p

Varje uppg är värd 20p. 3 förhör 10p, 3 läxor 10p

Bonus upp till 10p för smarta lösningar elr tilläggsfunktioner

5p avdrag per förseningsvecka

Närvaro

Jag använder mig av en närvarolista.

De som inte har deltagit på nån av de två första föreläsningarna blir borttagna från ASTA

<70% närvaro => begränsad klagomålsrätt

Upplägg - Hårdaste 10 dagarna i kursen



Lektion 1 - Kursinfo, verktyg & resurser, Intro till Databearbetning. My first python app	Läxa 1 ut
Lektion 2 - Python Moduler och Klasser, My second and third app. Läxa 1 hjälp?	Förhör 1 ut, Läxa 2 ut
Lektion 3 - Python Datastrukturer, Numpy & Matplotlib, Uppg 1 start	Förhör 1 in
Lektion 4 - Pandas, Uppg 1 forts	Läxa 2 ut, Uppg 1 ut
Lektion 5 - Visualisering, Webscraping & BeautifulSoup, Pandas, Uppg 2 start	Uppg 1 in, Uppg 2 ut, Uppg 3 ut
Lektion 6 - Förhör 2 (Numpy, Matplotlib & Pandas), Ljud och Bilder som data	Förhör 2 ut?
Lektion 7 - Inlämningsuppg 3 fortsättning, Övning med Bilder och Signaler	Uppg 2 in, Uppg 3 in, Uppg 4 ut
Lektion 8 - Inlämningsuppg 4. Kodande & Feedback, Glögg på cornern?	Uppg 4 in

Inlämningsuppg 1 - Feedback?

100k filmratings dataset med ratings och demografi

1. **Läs in användardata, ratingdata och filmdata** - Vilken info finns i vilken fil?
Skapa tre variabler och använd `dtypes()`, `head()` och `describe()` för att få en bättre översikt över variablerna.
 2. **Välj data** - Visa endast kolumnerna Kön, Ålder och Yrke av användarna. ***Gruppera ratings enligt film?**
 3. **Filtrera data** - Få insikt i samband genom att visa endast användare som är över 40 och män
 4. **Utforska data** - Visa medelåldern av användarna som är författare. Räkna medelrating per film. Top 10 filmer.
 5. **Kombinera dataFrames** - Visa medelratingen per användare. Snäll & Tuff? Yrken bland män och vice versa?
- 11/33 (24) inlämnade kl 9.30**- Hur många har inte hunnit vs hur många har inte kunnat?

Python in one slide?



Python quirks - Indentation styr koden, försiktigt med mellanslag! Kolontecken efter if och else, *and or not*

Python - GPP, bygga webbsidor, analysera data, koda verktyg # Kommentar, även `""" Multiline comment """`

Variabler - behöver endast ett namn, tolken känner igen typen `"Sträng" + str(int) + "."`

Strings - "Text" eller 'strängar' **Escape chars** med `\` för att skriva t.ex citattecken bland strängar.

Numror - Decimaltecken . | j för komplexa tal | int -> float -> complex. Tolk konv till bredare innan aritmetik.

Aritmetik - % modulo returnerar resten, // returnerar kvoten, ** fungerar som exponent. *Se upp!* $2^{1/2} = ?$

Booleans - = för tilldelning (assignment), == för utvärdering. Efter `str(True)` går variabeln inte att använda i logik!

If elif else - `raw_input("Mata in en sträng")`. (Error handling) med `try: except Error:` + if else för input validation

Interaktiv hjälp:

dir() - Se vilka moduler, objekt, klasser och metoder ni har.

help(someObject.someMethod()) - Få tilläggsinformation om objekt, metod eller funktion

someObject.someMethod? - Visa docstring

jupyter-notebook - tryck tab 1-4 gånger för att utöka information om det ni håller på att skriva just nu

Numpy & np.array - betydligt färre for loops

Matematiska operationer över listor

```
numpy_bmi_array = list_of_weights / list_of_heights ** 2    #Bara en data type i array!
```

Märk också skillnad mellan `pylist + pylist` #konkatenering `np_array + np_array` #aritmetik

Array of booleans:

```
numpy_bmi_array > 20 returnerar en list av booleans:      [False,False]
```

```
numpy_bmi_array[numpy_bmi_array > 20] returnerar:      [ 24.20, 21,24 ] # Praktiskt!
```

2D numpy arrays: En förbättrad version av list of lists `array[0,10] * array[2,:]`

`array[row][column]` eller `array[row,col]` t.ex `array[2,3:5]` # Fjärde och femte kolumnen på tredje raden.

Numpy simple data analytics `np.mean()`, `np.median()`, `np.std()`, `np.corrcoef()`, `np.column_stack()`

Om du delar upp datan i två np.arrays, nycklar och värden, kan du hänvisa till index med endast nyckelvärden

```
positions = ['GK', 'M', 'A', 'D', ...]      heights = [191, 184, 185, 180, ...]
```

```
gk_heights = heights[positions=='GK']      # Superhändigt!
```

Matplotlib - Visualisering made easy

Matplotlib - `import matplotlib.pyplot as plt` # Använder pyplot paketet från matplotlib

Line och pie `plt.plot(x,y)` # `kind='bar'` `kind='barh'` `kind='pie'`

Färger - **colormaps** - sekventiell, divergent, kvalitativ `plt.plot(colormap='Pastell')`

Scatter `plt.scatter(x,y)`

Skalor `plt.scale('log')`

Histogram - `plt.hist(data, bins=10)` Visualisera distribution av data # standard Python optional variable!

Anpassa graf - `plt.xlabel("X-axel")` `.title` `.yticks` # använd aritmetik för att förbättra det visuella meddelandet

Läs om flera alternativ för [pyplot.plot](#) och [pyplot.scatter](#)

%matplotlib inline - För att få matplotlib o funka inline i jupyter:

Idag - Matplotlib och visualisering

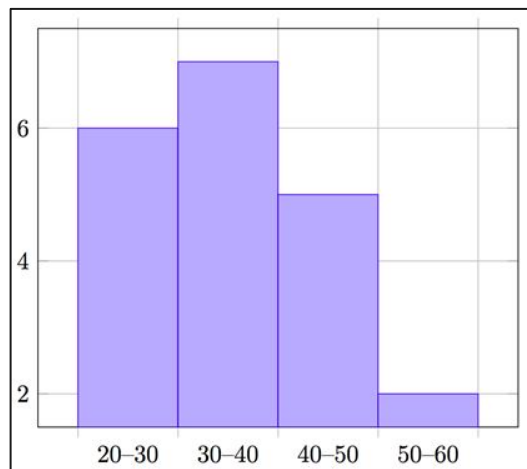
1. [Chapter 1](#) - DataCamp - *Doit*
2. [Picking the right graph for your data](#) - Olika grafer för olika data - *Sen*
3. [Basic Plotting](#) - MatPlotLib & Pandas
4. [Inline Plotting](#) - Charles Kelly *mwah super bass*
Avancerade grafer, flera linjer etc
5. [Chapter 2](#) - Line Bar and Pie Plots (även intro to seaborn) [Lilian Pearson!](#)
På djupet standardgrafer

Histogram - Hur är min data distribuerad?

Histogram - Används för att åskådliggöra en distribution med många värden

Exempel: Åldersfördelning, Längdfördelning

Ingen point att säga "Av 30 arbetare finns det 1 som är 20år, 2 som är 21år, 1 som är 22 år..."



Klasser - På x-axeln

Skapa de här med en for loop och en if sats

Frekvens - På y-axeln

Skapa en counter tabell som räknar hur många gånger vi faller inom klasserna.

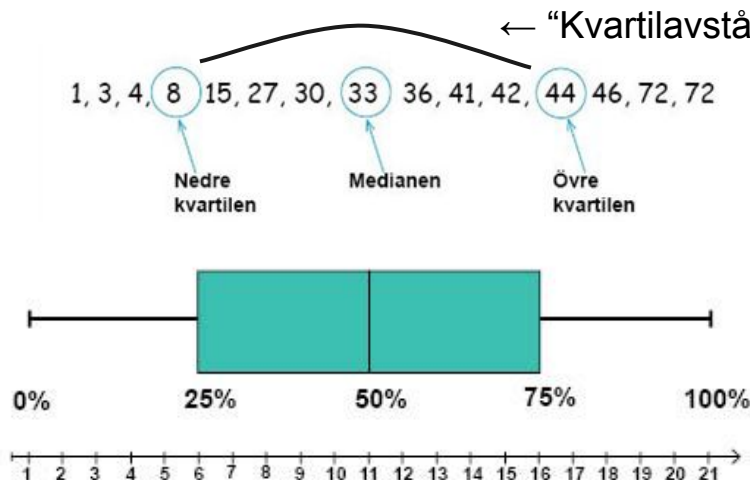
Exempel: `df.describe()` men visuellt!

[Matplotlib.pyplot.hist](#) **Uppg:** [Läs den här sidan](#)

Låddiagram - Are you a part of the 50th %ile

Låddiagram - Baserat på tre mått

1. Variationsbredd/Spridning: Maxvärde - minvärde
2. Kvartiler/Fjärdedelar: Beskriver spridningen kring medianen



3. Percentiler:

"Till P33 hör värden 1,3,4,8,15"

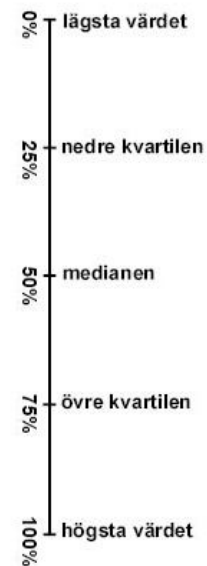
Samma som Kvartiler men helt annorlunda ;)

Q1 - P25 (alltså neråt!)

Q2 & Q3 - Medianen P50 (upp och ner!)

Q4 - P75 (alltså uppåt!)

Uppgift: Läs, ladda ner demofilen, lek med den
[Pyplot.subplots, axis.boxplot](#)



Visualisering



[Online Akademin - MatteCentrum](#) - Vad är ett histogram

[Online Akademin - MatteCentrum](#) - Vad är lådagram

[Matteboken - Kvartiler och Lådagram](#) - Statistik repetition

[Matteguiden - Spridningsmått, lådagram](#) - Statistik repetition

Vi lyssnar i klassen

[Tidsserier - Emma Saunders - Lynda](#) - Nu öronen på skaft!

[Distribution - Emma Saunders - Lynda](#) - Ska ja sätta på 0.75x?

Visualisering



[Online Akademin - MatteCentrum](#) - Vad är ett histogram

[Online Akademin - MatteCentrum](#) - Vad är lådagram

[Matteboken - Kvartiler och Lådagram](#) - Statistik repetition

[Matteguiden - Spridningsmått, lådagram](#) - Statistik repetition

Vi lyssnar i klassen

[Tidsserier - Emma Saunders - Lynda](#) - Nu öronen på skaft!

[Distribution - Emma Saunders - Lynda](#) - Ska ja sätta på 0.75x?

[Hierarisk Data - Emma Saunders - Lynda](#) - Kämpa! Bara en kvar!

Visualisering



[Online Akademin - MatteCentrum](#) - Vad är ett histogram

[Online Akademin - MatteCentrum](#) - Vad är lådagram

[Matteboken - Kvartiler och Lådagram](#) - Statistik repetition

[Matteguiden - Spridningsmått, lådagram](#) - Statistik repetition

Vi lyssnar i klassen

[Tidsserier - Emma Saunders - Lynda](#) - Nu öronen på skaft!

[Distribution - Emma Saunders - Lynda](#) - Ska ja sätta på 0.75x?

[Hierarisk Data - Emma Saunders - Lynda](#) - Kämpa! Bara en kvar!

[Geografisk och Text - Emma Saunders - Lynda](#) - Okej två men dom e korta ^^

Pandas - Notes från Fernandes & Kelly Ch2



Dataframe - 2D array like from numpy

Series - 1d array of indexed data (col 1)

DataFrame['Series'] - Access series in dataframe. #Different functions for DF and Series, do type(obj)

DataFrame[['series1', 'series2']] - Access several series from dataframe # Märk att svaret är en ny DF

Data Input - Stöd för read_csv, read_excel, read_json, read_sql_table

DataFrame.shape - tuple for confirming dataframe dimensions

DataFrame.head() and **DataFrame.tail()** - visar första eller sista raderna från DF, mycket praktiskt

DataFrame.info() översikt för DF, märk datatyper!

DataFrame.describe() ger dig counts och mean min max quartiles

DataFrame.T står för transpose och gör kolumner till rader

DataFrame.loc[:,['A','B']] - Index är tillåtet, pandas förstår också sig på datum, **loc()** för att välja enligt label

[inLearning Pandas Selection](#) 5 min framåt

Pandas - Kelly Ch2 bredare & djupare än Fernandes



DataFrame - 2D array like from numpy

Series - 1d array of indexed data (column) # en 1D DF "med en col" ser ut som en rad, don't be fooled

DataFrame['Series'] - Access series in dataframe. # Different functions for DF and Series, do type(obj)

DataFrame[['series1', 'series2']] - Access several series from dataframe # Märk att svaret är en ny DF

Data Input - Stöd för read_csv, read_excel, read_json, read_sql_table

DataFrame.shape - tuple for confirming dataframe dimensions

DataFrame.head() and **DataFrame.tail()** - visar första eller sista raderna från DF, mycket praktiskt

DataFrame.info() översikt för DF, märk datatyper!

DataFrame.describe() ger dig counts och mean min max quartiles

DataFrame.T står för transpose och gör kolumner till rader

DataFrame.loc[:,['A','B']] - Index är tillåtet, pandas förstår också sig på datum, **loc()** för att välja rad enligt label

[inLearning Pandas Selection](#) 5 min framåt

Data analysis - Lite praktiska metoder



df.year.value_counts(dropna=false) - hur många värden i fallande ordning #hur många filmer per år sort desc

df.sort_values(by=['rating','title'], inplace=false) - sortera enligt rating, sedan filmtitel, skriv inte över

df[(df.oscar >= 1) & (df.rating >= 3.4)] - Boolean indexing #Visa endast top rated oscarfilmer

- Flera krav inom parenteser, and operand & (visst minns ni?)

df.str.genre.contains("Horror") - startswith(), isnumeric() # Visa horror filmer

Querying Data Frames - Ett par övningar ([Manipulating DF w Pandas](#) [Pandas Foundations](#))

Ex:

```
filtered = df[ (df.genre == "Horror") & (df.oscar == 1)] # Visa horrorfilmer som fått oscars
```

```
filtered.sort_values('oscar', ascending=false) # sort by oscars desc.
```

```
filtered[ ['rating', 'title'] ] #Visa bara 2 col
```


BeautifulSoup - Senast: Data Scraping



Vilka produkter finns på newEgg?

Exempel: Vi gör tillsammans ^.^

my_url =

"<https://www.newegg.com/Video-Cards-Video-Devices/Category/ID-38?Tpk=graphics%20card>"

Vad bra ni som följde med! Det blir Läxa #2!

För kommande Scrapinguppgift vore det bra att:

Läsa: [Intro till BeautifulSoup](#)

Sedan: Läs [dokumentationen](#)

BeautifulSoup - Intro till Data Scraping

Senast: hitta grafikortens element på newegg.com **Nu:** Samma grej för steam sale sajten?

```
In [13]: from bs4 import BeautifulSoup  # Import BeautifulSoup
from urllib.request import urlopen  # Import urlopen
soup = BeautifulSoup(urlopen('http://store.steampowered.com').read()) #make soup
containers = soup.findAll("div", {"class": "discount_final_price"}) #findALL containers
print("Sale Price: " + containers[0].text)
```

Sale Price: 4,99€

List comprehension: for loop och list creation oneliner

```
[t["class"] for t in soup.find_all("table") if t.get("class")]
```

Lambda uttryck: returnerar värde av uttrycket inom sig

```
rem_nl = lambda s: s.replace("\n", " ")
```

BeautifulSoup - Bra snabbstartresurser



1. [Intro to BeautifulSoup](#) - 10 min quickstart ifall ni missa newegg
2. [Intro till BeautifulSoup \(Text\)](#) - Python for beginners
3. [Docs för BeautifulSoup](#)

Sen då?



Nytt försök på Labb 2 - Numpy, Matplotlib och Pandas

Labb 2 från Harvards kurs CS109 Data Science

<http://cs109.github.io/2015/pages/videos.html>


Övning:

Finlands befolkningsutveckling - Scrapea [wikipedia](#) och rita barchart

Inlämningsuppg 2

Steam Sales? <https://store.steampowered.com/search/?specials=1&os=win>

Lynda och resurser - Kolla även itslearning!

- 
- Cheat sheet: [Anaconda Cheat Sheet - Getting Started - PDF](#)
[Pandas Cheat Sheet - PDF](#)
- Manual/Docs [Conda package manger - Docs](#)
[Pandas - QuickStart & Cookbook](#)
- Tutorials (text) [Anaconda Getting Started - User Guide](#)
[Python - Intro till avancerat - Övningar och förklaringar](#)
[Pandas tutorial - PythonSpot](#)
[Intro to data science Numpy, MatPlot & Panda](#)
[\(Pandas - How do pivot tables work - ExcelCampus\)](#)
- Tutorials (video) [Socratica python tutorial - Youtube](#)
[Derek Banas - "Learn Python in one video"](#)

Lynda och resurser2 - Kolla även itslearning!



Interaktiva:

[Intro to **python** for data science - Gratiskurs - DataCamp](#)

[Intro to python for data science - Ch4 - **Numpy** \(DataCamp\)](#)

[Intermediate python for data science - Ch1 - **MatPlotLib** \(DataCamp\)](#)

Lynda:

[6h nybörjarkurs **Python** för datavetenskap med Lillian Pierson - Lynda](#)

[2h intermediate - **Numpy** Data Science Essentials - Charles Kelly](#)

[Intermediate - Ch3: **Numpy**, Ch4: **Pandas**, Ch 9: **matplotlib** - Miki Tebaka](#)

[2h intermediate kurs i **Pandas** med Jonathan Fernandes - Lynda](#)

[2h intermediate **Pandas** för Datavetenskap med Charles Kelly - Lynda](#)

[Big Data Analysis in python using **Numpy** and **Pandas** - Michele Vallisneri](#)

Lynda och resurser3 - Kolla även itslearning!



Interaktiva roligheter

[The Python Challenge](#)

[Roliga övningar i logisk ordning - Practice Python](#)

[How to think like a Computer Scientist](#)

[CodeSignal - Interaktiva utmaningar, badges, points etc.](#)

[Reddit daily programmer challenges](#)

Vill du vinna 1 miljon \$

[7h gratis tutorial på Kaggle - Känner ni till kaggle?](#)

Interaktiva, bra helheter

Python 2 vs Python 3 trubbel, kolla [här](#)

Jag har redan fallit av kälken! - Brush up ur skills



1. Kolla [Socratica tutorialen](#) videorna 1-17
2. [Chapter 1-3 - Python for data science \(DataCamp\)](#)
3. [Chapter 1 - Writing python functions \(DataCamp\)](#)
4. [Chapter 4 - Numpy \(DataCamp\)](#)
5. [Chapter 1 - Matplotlib \(DataCamp\)](#)
6. [Chapter 1-4 - Pandas for data science \(Lynda: Kelly\)](#)

Sök hjälp bland resurserna om du kör fast.

Tidigare Läxor: Förhör på innehållet coming soon?

Step 1:

[Intro to python for data science](#)
[Chapter 4 - Numpy \(DataCamp\)](#)

Step 2:

[Intermediate python for data science](#)
[Chapter 1 - Matplotlib \(DataCamp\)](#)

Step 3:

[Pandas Essential Training -> Kapitel 6 \(Fernandes\)](#) - Ytlig?
[Pandas for data science -> Kapitel 5 \(Kelly\)](#) - Långsam?

Step 4:

[Data Exploration, Distribution analysis,](#)
[Categorical variable analysis, Data Munging](#)

Hur långt har ni kommit? Ointressanta resurser är även mitt problem..

Python refresher?



1. [Intro to Python for data Science](#) - Python och Numpy
Ganska basics men nog bra (Gratiskurs)
2. [Intermediate Python for Data Science](#) - Matplotlib & Pandas
Chapter 2 riktigt bra!

Numpy refresher



1. [Chapter 4](#) - DataCamp
2. [DataFrames and Panels](#) - Matplotlib & Pandas
3. [Chapter 3](#) - Numpy Arithmetics