

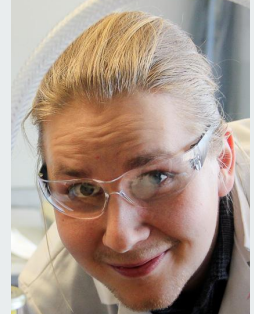


# Databearbetning

Steget innan datavetenskap

Lektion 7 - Inlämningsuppg 3 och ljud som data

Dennis Biström  
bistromd@arcada.fi



# Upppg 3 - 28.10



## 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>

# Upppg 4 - 4.11

## Bedömning

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

Varje uppg är värd 20p. *1 förhör 10p, 2 läxor 15p*

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 - Vi e på slutrakan



**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** - Visualisering forts. Matplotlib med textfiler, Ljud och Bilder som data

**Uppg 2 in 21.10 kl 16.59**

**Lektion 7** - Övning med Ljud & kodande på inlämningsuppg 3

**Uppg 3 in 28.10 kl 16.59**

**Lektion 8** - Övning med bild & kodande på inlämningsuppg 4

**Uppg 4 in 4.11 kl 16.59**

Feedback & Glögg på cornern? 1.11 elr 8.11?

# Inlämningsuppg 2 - Feedback?

**Steam Sale!** BeautifulSoup och Pandas

Ladda ner erbjudanden på steam. Skapa en tabell med följande rubriker och fyll tabellen med data från sajten.

Spelnamn|Rating|#Reviews|Rabatt%|Pris|OrdinariePris|Utgivningsår|Win|Lin|OSX|Tid

Skapa en sifferbaserad rating enligt alternativen som steam använder i textform (Mostly Positive = 4/7...).

Använd 0 eller 1 för att beteckna ifall spelet stöder platformen (operativsystem).

För fulla poäng krävs att man samlar data från 5 sidor med erbjudanden.

Skapa en CSV fil (men endast ifall den inte redan finns!) och skriv in datan som har samlats.

På så vis kan scriptet skapa en fil ifall den inte finns, och ifall den finns fortsätter datainsamlingen.

**18/33 inlämnade, ja e nöjd! Proj 1 hade 20**

Hur många har inte hunnit vs hur många har inte kunnat?

# Senast - Visualisering

**Andrew Gelman - Why tables are better than charts** (Skriven första april)

“Graphs are a way of implying results that are often not statistically significant”

En välstrukturerad tabell är ärlig.

Vi är vana med tabulär data och har inga problem att uppfatta förhållanden. Använd för presentation

Alla grafer uppmuntrar eller insinuerar till slutsatser. Använd för att övertyga

## **Gross National Happiness i Bhutan**

40.8% of people in Bhutan have achieved happiness.

The GNH Index requires an array of conditions to be met.

Those who are happy enjoy it in 56.6% of the domains.

Happiness (GNH) is reached when people reach sufficiency in roughly half of the domains.

Källa: [World Happiness Report \(2012\)](#)

[Årets rapport](#)



# Visualisering

## Andrew Gelman - Why tables are better than charts (Skriven första april)

“Graphs are a way of implying results that are often not statistically significant”

En välstrukturerad tabell är ärlig.

Vi är vana med tabulär data och har inga problem att uppfatta förhållanden. Använd för presentation

Alla grafer uppmuntrar eller insinuerar till slutsatser. Använd för att övertyga

Figure 11: GNH index by gender

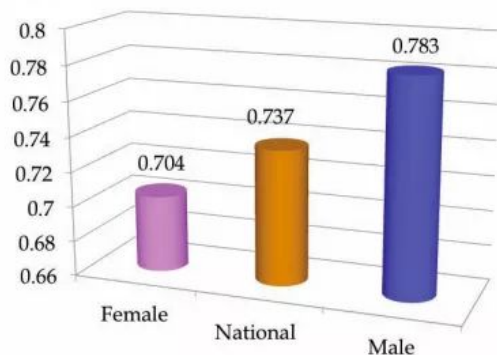


Figure 11

When we decompose the GNH Index by gender we see that men are happier than women.

49% of men are happy, while only one-third of women are happy.

Gender	Happiness
Men	49 %
Women	33 %

# Tabellen

**Remove**  
to improve  
the **data tables** edition

# Visualisering

Andrew Gelman - Why tables are better than charts (Skriven första april)

“Graphs are a way of implying results that are often not statistically significant”

En välstrukturerad tabell är ärlig.

Vi är vana med tabulär data och har inga problem att uppfatta förhållanden. Använd för presentation

Alla grafer uppmuntrar eller insinuerar till slutsatser. Använd för att övertyga

Figure 4: In which domains do happy people enjoy sufficiency?

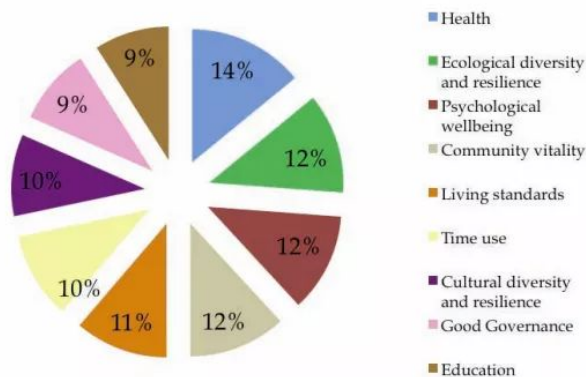


Figure 4

Shows in which domains happy people enjoy sufficiency.

We can see that all nine dimensions contribute to GNH

Happy people live relatively balanced lives.



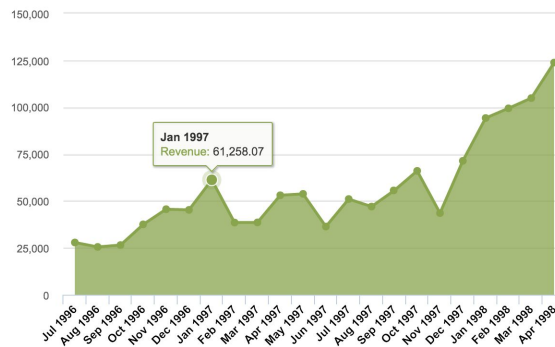
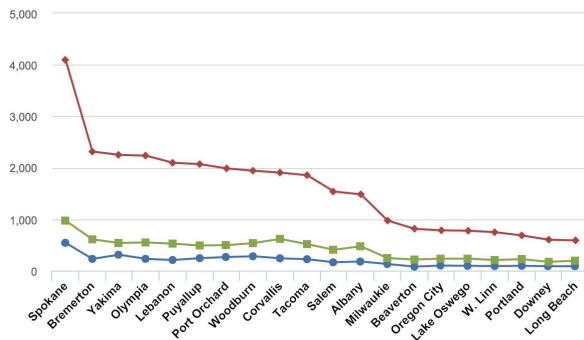
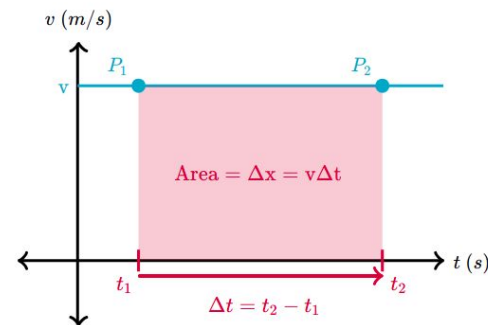
# Pajdiagram

**Remove**  
to improve  
the **pie chart** edition

# Vilken graf - Kontinuerlig data

1. Är det en tidsserie du vill visa?
  - a. Är det numerisk data?
    - i. **Kontinuerlig data?** Ex temperatur

Linje eller Area - Fördelen med area är att integralen kan ge mervärde



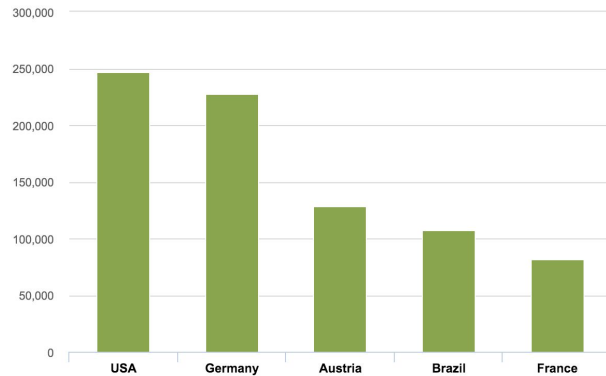
# Vilken graf - Diskret data

1. Är det en tidsserie du vill visa?

a. Är det numerisk data?

i. **Diskret data?** Försiktigt med sampel och medeltal

**Barchart eller Candlestick** - Fördelen med candlestick är att man kan illustrera variation



Och förutspå framtid?

# Barchart

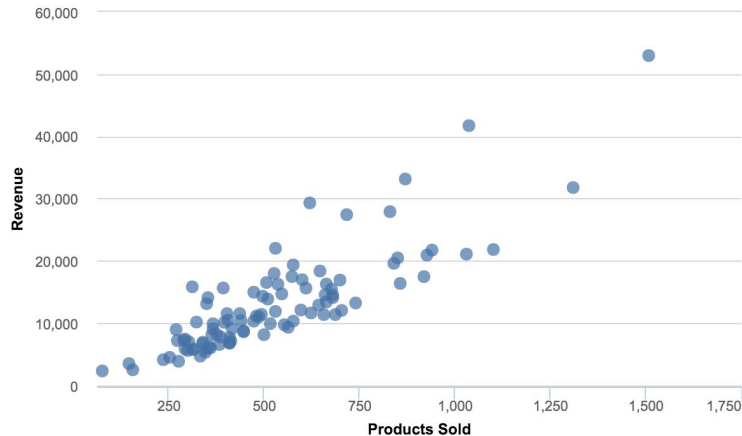
**Remove**  
to improve  
(the **data-ink** ratio)

# Vilken graf - Korrelation

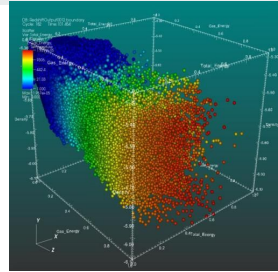
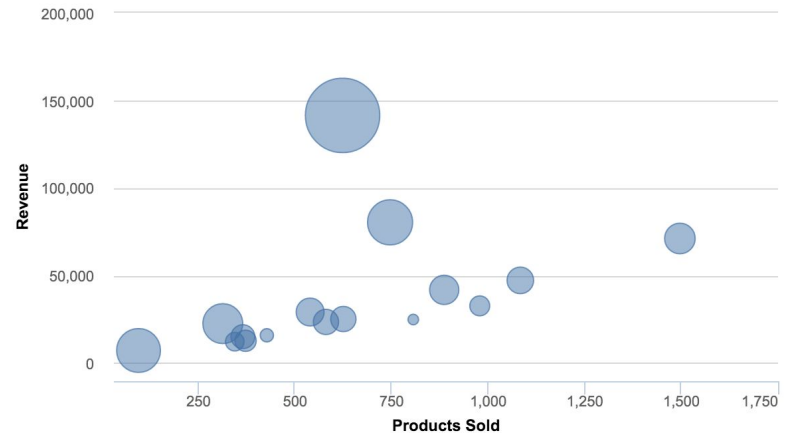
2. För korrelation och distribution (men även det snabbaste sättet att få en insikt i din data)

**Scatter** aka punktdiagram, även spridningsdiagram eller sambandsdiagram

Illustrerar även outliers eller gruppering



Få en insikt i samband mellan upp till fyra variabler



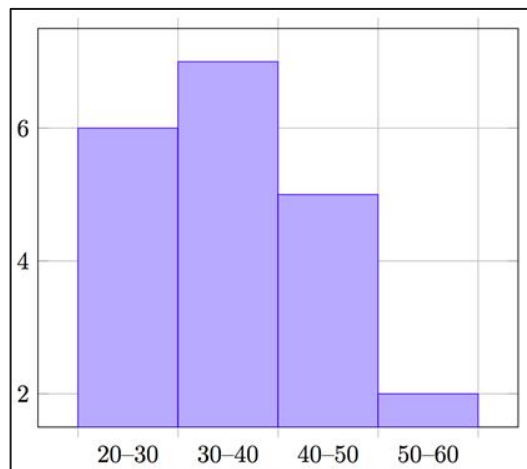
# Histogram - Distribution

(även för fler än 7 staplar)

**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](#)    **Läxa:** [Läs den här sidan](#)

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

## Låddiagram - Baserat på tre mått

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

“Kvartilavstånd = 50% av värden →

1. Percentiler:

Till P33 hör värden 1,3,4,8,15

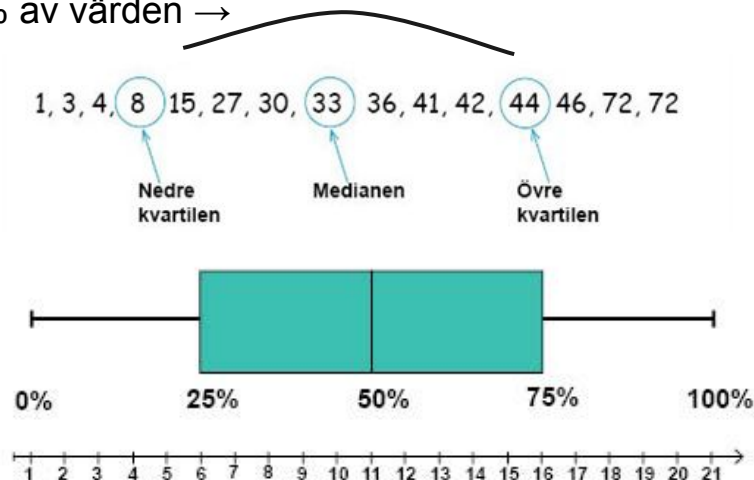
Q1 - P25 (alltså neråt!)

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

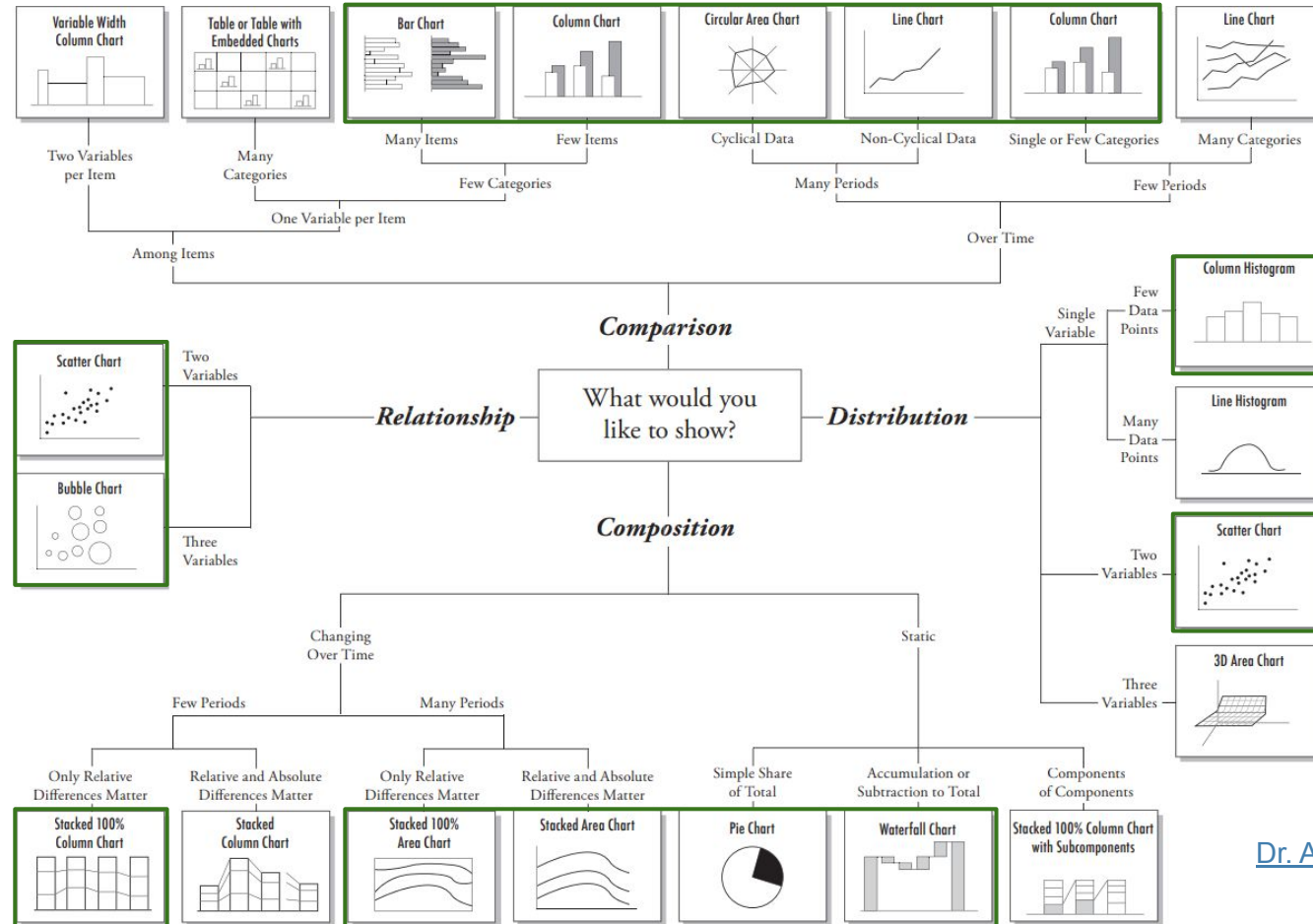
Q4 - P75 (alltså uppåt!)

**Läxa: Läs, ladda ner demofilen, lek med den**

[Pyplot.subplots, axis.boxplot](#)



# The Chart Chooser och The Slide Chooser

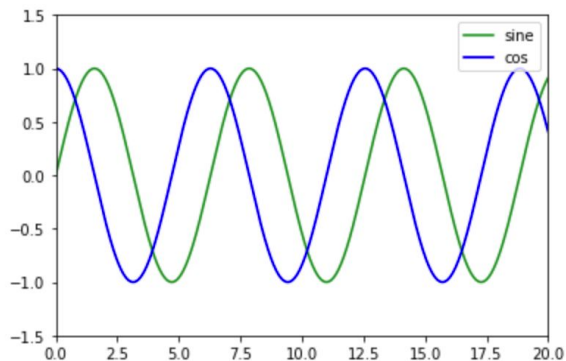


Dr. Andrew Abela

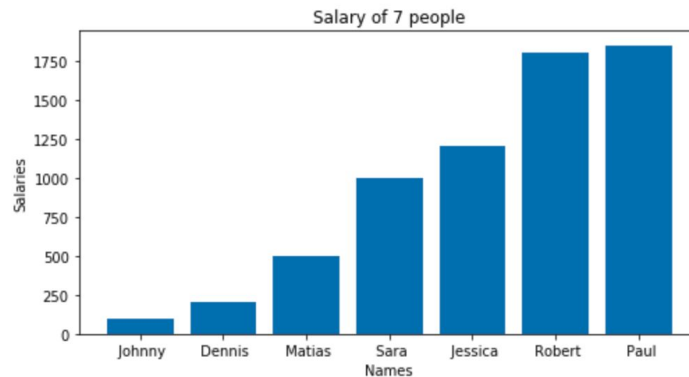


# Senast - Matplotlib och Numpy

```
# Sin och Cos med legend & limits|
plt.plot(x , y1, "-g", label="sine")
plt.plot(x , y2, "-b", label="cos")
plt.legend(loc="upper right")
plt.ylim(-1.5, 1.5)
plt.xlim(0, 20)
plt.show()
```



```
# Vi använder indexing för att få bort stuff
plt.figure(figsize=(8, 4))
plt.bar(x[1:-2], salary[1:-2])
plt.xticks(x[1:-2], names[1:-2])
plt.ylabel("Salaries")
plt.xlabel("Names")
plt.title("Salary of 7 people")
plt.show()
```



```
# Märk hur median vs mean ändrar
print('\nMedel: ', np.average(salary[1:-2]), "\nMedian: ", np.median(salary[1:-2]))
```

# Ljud - Vi är påväg mot bilder som data



**Vågformer** - Ljus och ljud är vågformer\*

Många signaler behöver ett medium att färdas i, ljud är skillnad i lufttryck

En våglängd är avståndet mellan toppar/dalar. Våglängd betecknas lambda  $\lambda$

I en kontinuerlig signal bestämmer våglängden frekvensen  $f$

Vi mäter ljudfrekvens i enheten Hz som betyder 1/s "hur många svängar per s"

**Människan** hör 20-20kHz ljud, men vad ser vi för våglängder?

Kan man se ljud?

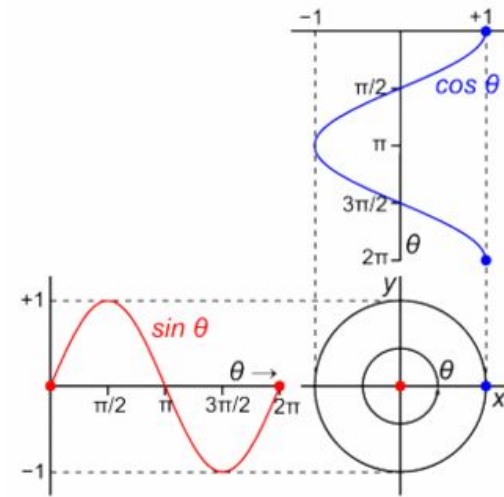
Kan man höra hastighet?

**Amplitud** - Hur stora värden sparar vi?

**Period** - Vad är frekvensen eller våglängden?

**Fas** - Börjar vi från noll?

Hör människan fasförskjutning?



# Ljud - Vi är påväg mot bilder som data

**Vågformer** - Ljus och ljud är vågformer\*

Många signaler behöver ett medium att färdas i ljud är skillnad i lufttryck

En våglängd är avståndet mellan toppar/dalar. Våglängd betecknas  $\lambda$

I en kontinuerlig signal bestämmer våglängden frekvensen  $f$

Vi mäter ljudfrekvens i enheten Hz som betyder 1/s "hur många svängar per s"

**Människan** hör 20-20kHz ljud, men vad ser vi för våglängder? 430-770 THz

Kan man se ljud? [Refraktion](#) Kan man höra hastighet? [Dopplereffekt](#)

**Amplitud** - Hur stora värden sparar vi?

**Period** - Vad är frekvensen eller våglängden?

**Fas** - Börjar vi från noll?

Hör människan fasförskjutning? Nej



[Source](#)



# Sampling - Riktiga världen och datorn

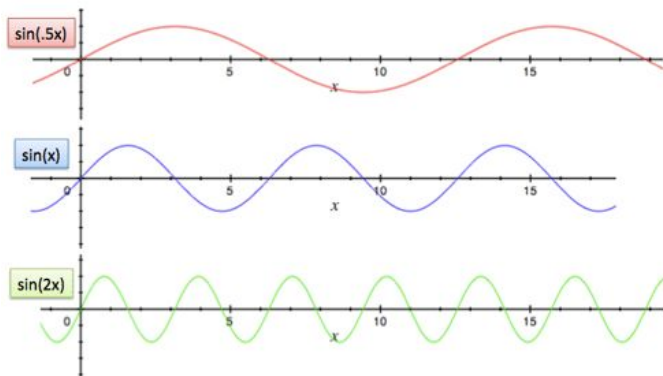
**Vågformer** - Ljus och ljud är vågformer\*

Alla signaler och all data går att uttrycka med sinusvågor.

Det är inte helt omöjligt att spara sinusvågor på en dator, men nästan. ([wavelets](#))

Därför SAMPLAR vi data och sparar diskreta värden.

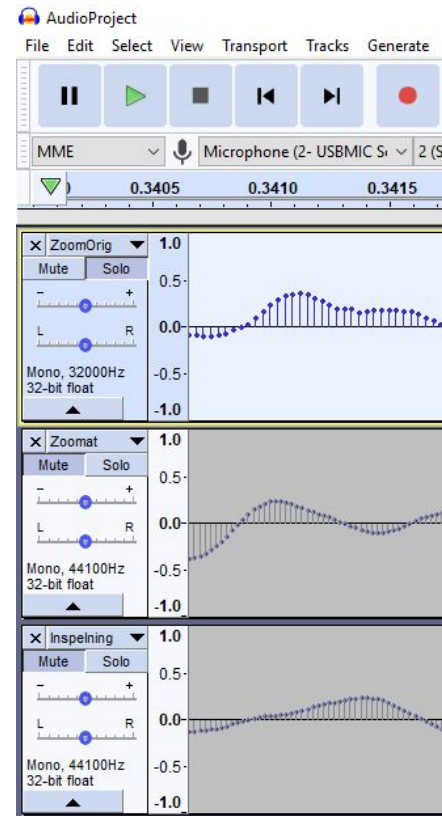
**Exempel** - Octave och Dataset



Mono  
32kHz  
32-bit float?

Varför 44100Hz  
aka "CD-Quality"

Nyquist frekvens



# Sampling - Nyquist

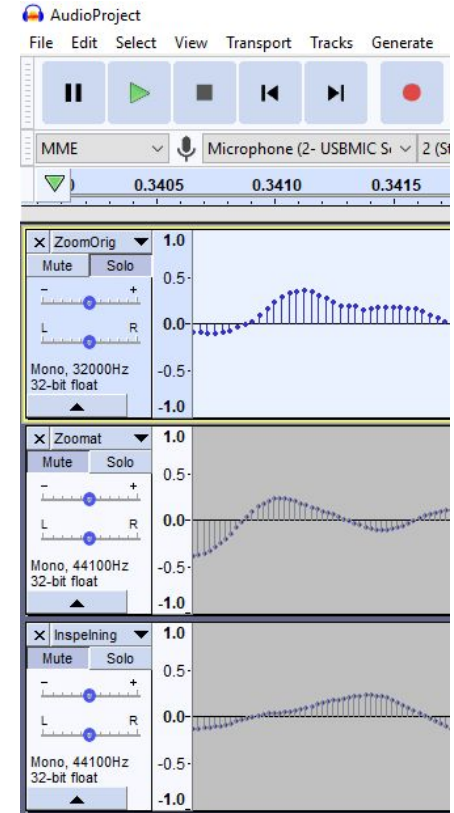
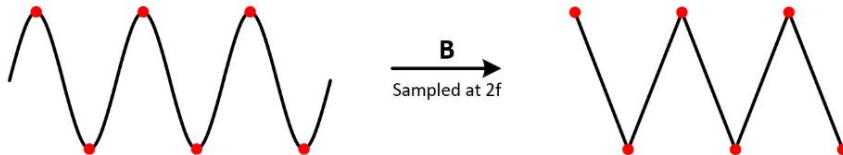
**Sampl frekvensen = tonens svängningsfrekvens**

(A = 440hz, 440 datapunkter per sekund)



**Sampl frekvensen = 2 ggr tonens svängningsfrekvens**

(A = 440hz, 880 datapunkter per sekund) - "Good enough"



# Inlämningsuppg 4 - Deluppg 1



## Inlämningsuppg 4 - Deluppg 1

Skapa tre sinusvågor med numpy i jupyter notebook, gärna ett vackert [C-accord](#)

Minns att ljudsampler per sekund definierar vad vi kallar sample rate

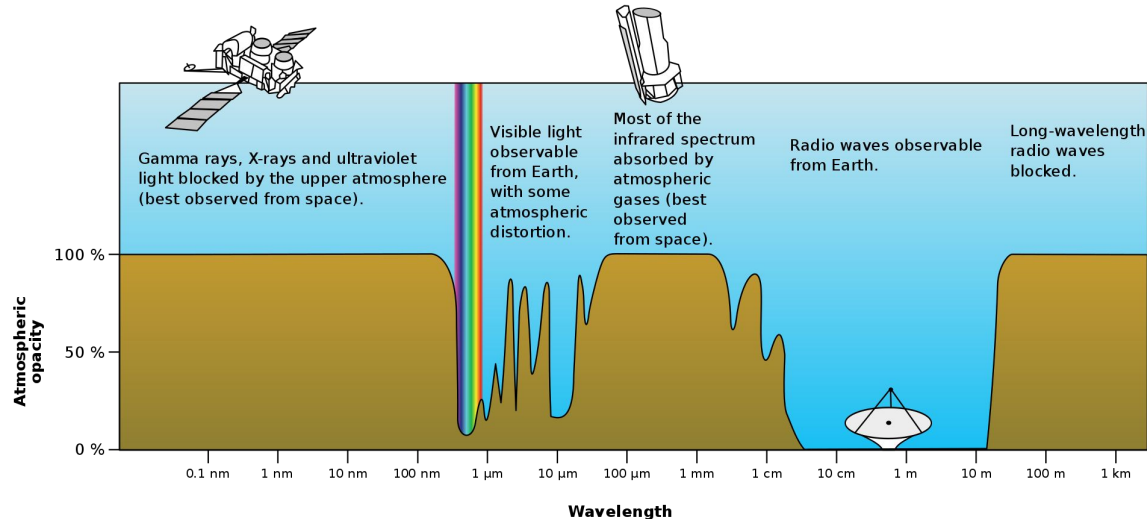
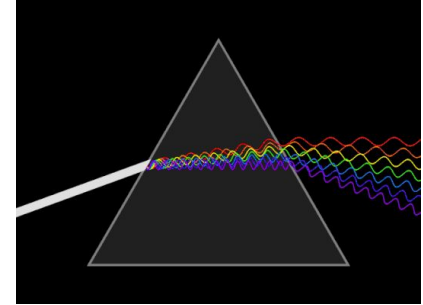
Märk att längden av np.array måste vara minst dubbelt sampelns frekvens för att du ska kunna skapa en ton

# Ljus - EM strålning

**Vågformer** - Ljus och ljud är vågformer\*

Synligt ljus ligger vid 400 - 700 nm ,dvs frekvenser vid 430-770 THz

Vi har valt att spara färg digitalt i rutor (pixlar) med tre färgkomponenter RGB



# Ljus - EM strålning

**Vågformer** - Ljus och ljud är vågformer\*

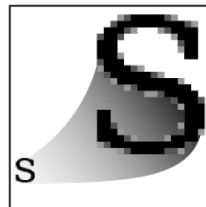
24-bitars färgrymden reserverar 8 bitar per kanal RGB

Det här ger oss  $2^{24} = 16,777,216$  färger

På nätet uttrycker vi färger med hexadecimal, dvs tecken från 00-FF istället för 0-9

 rgb(255,0,255) blir #FF00FF

Blu-Ray och DVD standarden använder YCbCr eftersom det möjliggör lättare kompression av färgkanalerna i jämförelse med RGB. (Lossless conv. till RGB)



**Raster**  
.jpeg .gif .png





# Inlämningsuppg 3 - Vi börjar tillsammans



## Inlämningsuppg 3

Moore's lag säger att antalet transistorer i en mikroprocessor fördubblas ungefär varje 2 år.

[Moore's Law over 120 Years](#)

Stämmer det?

Ta in data från [https://en.wikipedia.org/wiki/Transistor\\_count](https://en.wikipedia.org/wiki/Transistor_count) om år och transistorantal

(OBS! scraping, tvättning, sorting av data behövs som vanligt), och rita tre grafer av transistorantalens utveckling.

1. Stapeldiagram CPU:s/Decade
2. Linjediagram nm:s/Decade
3. Scatter transistors/year\*

\*Använd logaritmisk skala på y-axeln, och sätt in en linje som visar vad ökningen borde vara enligt Moores lag.

Märk ut några valda punkter med pro세서orns namn.

# 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.


# Läxor:



- 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..***

# 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](#)