

# Programming

## Numerical Data Analysis & Visualization

Luna Pianesi

Faculty of Technology, Bielefeld University

```
332
333
334     if extrapolate is None:
335         extrapolate = self.extrapolate
336         x = np.asarray(x)
337         x_shape, x_ndim = x.shape, x.ndim
338         x = np.ascontiguousarray(x.ravel(), dtype=np
339
340     # With periodic extrapolation we map x to the
341     # [self.t[k], self.t[n]].
342     if extrapolate == 'periodic':
343         n = self.t.size - self.k - 1
344         x = self.t[self.k] + (x - self.t[self.k]) *
345         extrapolate = False
346
347     out = np.empty((len(x), prod(self.c.shape[1:])),
348                   dtype=self._ensure_c_contiguous())
349     self._evaluate(x, nu, extrapolate, out)
350     out = out.reshape(x_shape + self.c.shape[1:])
351     if self.axis != 0:
352         # transpose to move the calculated values to t
353         l = list(range(out.ndim))
354         l = l[x_ndim:x_ndim+self.axis] + l[:x_ndim] + l[x_ndim+self.axis:]
355         out = out.transpose(l)
356     return out
357
358     def _evaluate(self, xp, nu, extrapolate, out):
359         _bspl.evaluate_spline(self.t, self.c, reshape(self.c,
360         self.k, xp, nu, extrapolate, out)
361
362     def _ensure_c_contiguous(self):
363         """
364         c and t may be modified by the user. The Cython code
365         ensures that they are C contiguous.
366         """
367         self.c = np.ascontiguousarray(self.c)
368         self.t = np.ascontiguousarray(self.t)
```

# *Recap*

# Reading from file

Dynamic: read from file with name requested by prompt

```
1 fName = input('Input file: ')
2 lines = list()
3 f = open(fName)
4 for line in f:
5     lines.append(line)
```

# *File formats*

## *Unstructured data*

- Text

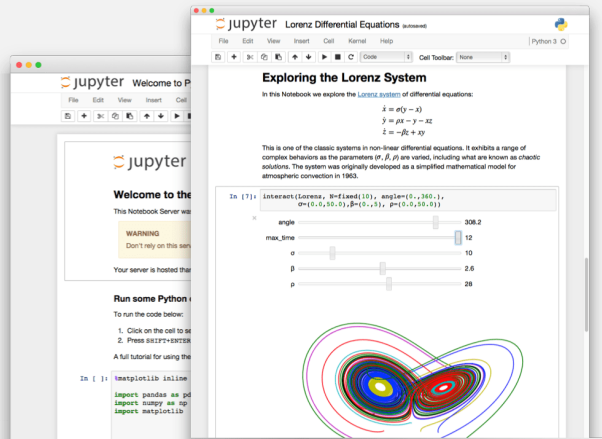
## *Structured data*

- XML
- JSON
- Tables
- Matrices

# Jupyter Notebook

## Why use Jupyter Notebook in Data Science?

- Simultaneous documentation & analysis
- Step-by-step processing
- Ensures reproducibility

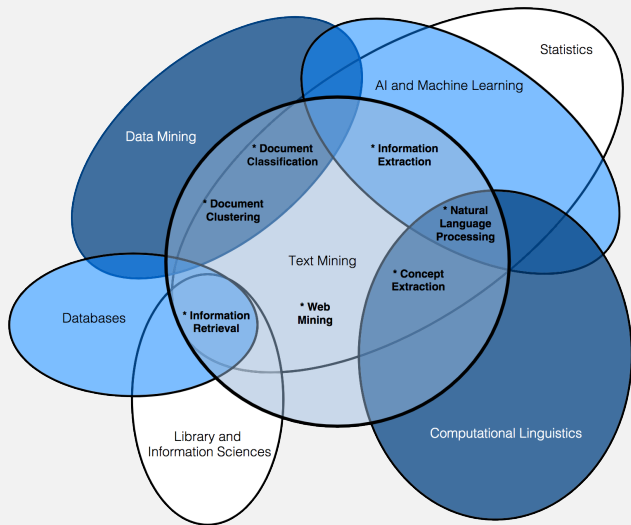


# Text mining

Relies on *Natural Language Processing (NLP)*

Main (constitutive) tasks:

- ❖ Document summarization, clustering & classification
- ❖ Information extraction
- ❖ Information discovery



source: Miner, Gary. *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. 1st ed. Amsterdam: Academic Press, 2012.

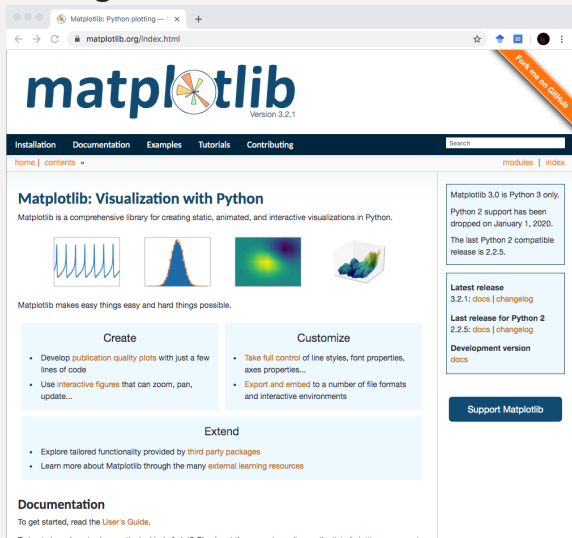
***Data  
Visualization***

***Numerical  
Data Analysis  
with NumPy***

***Modeling  
Experimental  
Data***

# Matplotlib: Visualization with Python

- ❖ de-facto standard library for scientific visualizations
- ❖ many third party packages built on top of Matplotlib
- ❖ comprehensive library for creating static, animated, and interactive visualizations



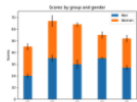
The screenshot shows the Matplotlib website homepage. At the top, there's a navigation bar with links for Installation, Documentation, Examples, Tutorials, and Contributing. Below the navigation bar, the main heading reads "Matplotlib: Visualization with Python" with the version number "Version 3.2.1". The page features several visualizations: a line plot, a histogram, a heatmap, and a 3D surface plot. The text describes Matplotlib as a comprehensive library for creating static, animated, and interactive visualizations in Python. It highlights key features under "Create" (developing publication quality plots, using interactive figures) and "Customize" (taking full control of line styles, font properties, axes, and exporting to various file formats). There are also sections for "Extend" (exploring tailored functionality and learning resources) and "Documentation" (getting started with the User's Guide). A sidebar on the right provides information about Python 3.0 support, the latest release (3.2.1), and the last release for Python 2 (2.2.5).

source: <https://matplotlib.org/>

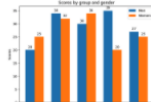


# Matplotlib: Visualization with Python

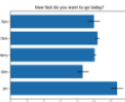
## Lines, bars and markers



Stacked bar chart



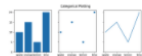
Grouped bar chart  
with labels



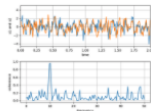
Horizontal bar chart



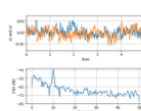
Broken Barh



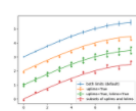
Plotting categorical  
variables



Plotting the  
coherence of two  
signals



CSD Demo



Errorbar limit  
selection

source: <https://matplotlib.org/gallery/index.html>

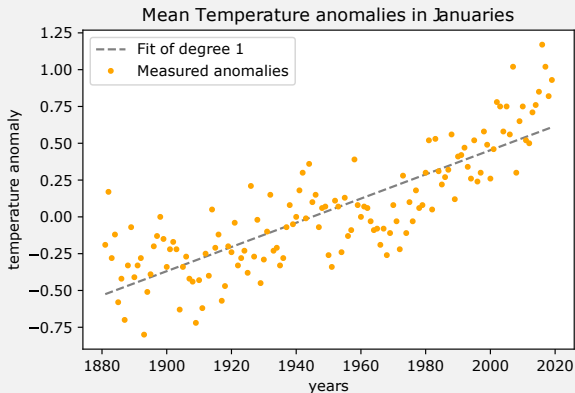
***Data  
Visualization***

***Numerical  
Data Analysis  
with NumPy***

***Modeling  
Experimental  
Data***

# Numerical analysis

- Numerical data: anything measurable
- Methods:
  - Interpolation and regression
  - Solving differential equations
  - Optimization
  - ...



# ***N-dimensional array:*** `numpy.ndarray`

## Array data structure

- ❖ immutable
- ❖ n-dimensional
- ❖ very storage efficient
- ❖ can store only data of same type

***Data  
Visualization***

***Numerical  
Data Analysis  
with NumPy***

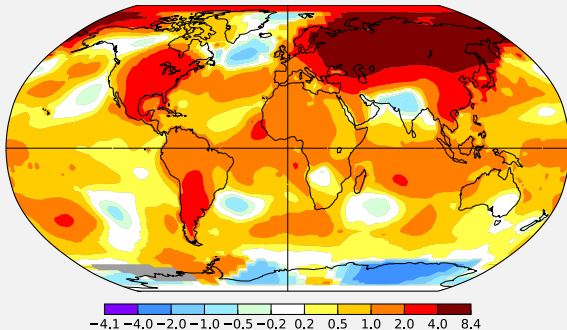
***Modeling  
Experimental  
Data***

# NASA's GISS Surface Temperature Analysis

March 2020

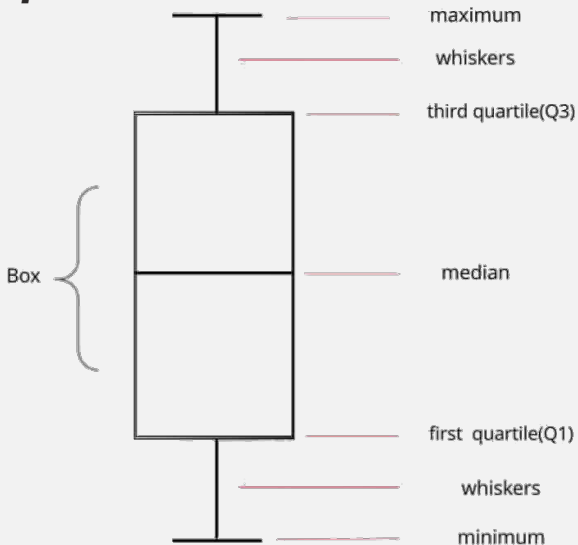
L-OTI(°C) Anomaly vs 1951-1980

1.18



- ❖ <https://data.giss.nasa.gov/gistemp>
- ❖ Collection of temperature data from thousands of meteorological stations
- ❖ Data represents *anomalies*, i.e., deviations from mean temperature measured in 1951-1980

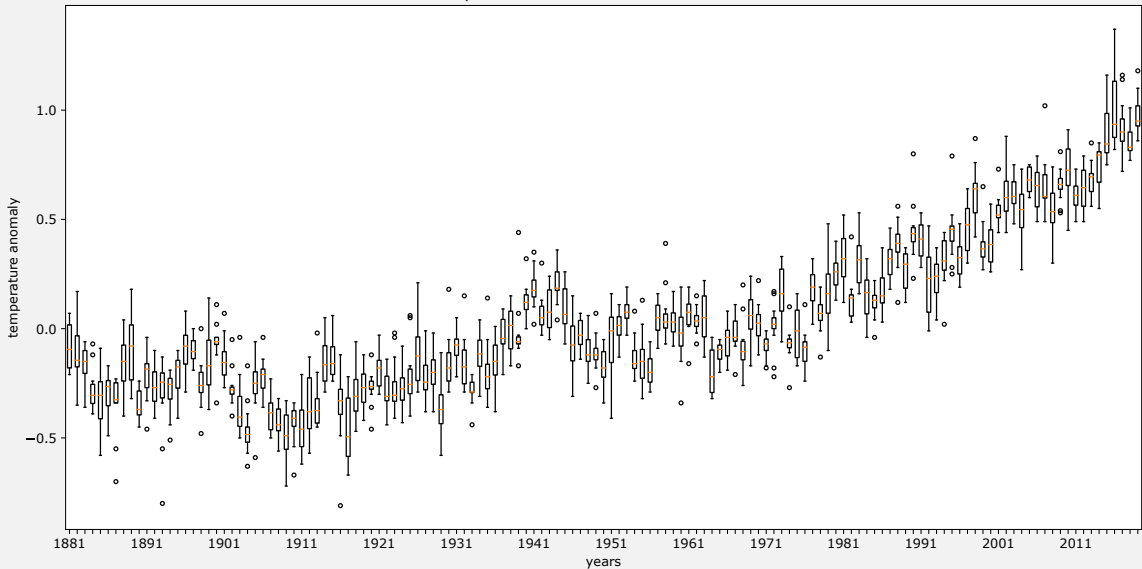
# Box (whisker) plot



source: [https://en.wikipedia.org/wiki/Box\\_plot](https://en.wikipedia.org/wiki/Box_plot)

# Whisker plot of GISS data

Temperature anomalies between 1881-2019





## ***Further material on methods in Data Science***

MIT Course 6.0002, Lectures on Understanding Experimental Data:

- <https://www.youtube.com/v/vIFKGF11Cn8>
- <https://www.youtube.com/v/fQvg-hh9dUw>

## Linear regression

Linear regression is a *linear* approach for modelling a predictive relationship between some parameters and a given input:

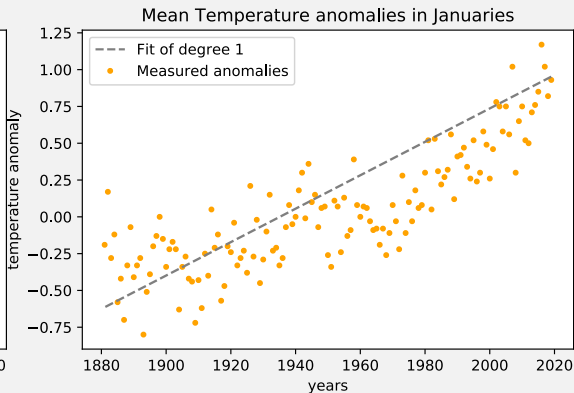
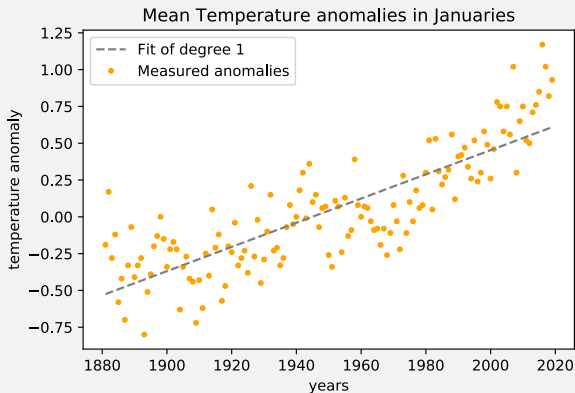
$$X = \begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{pmatrix}, Y = \begin{pmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{N-1} \end{pmatrix} \rightarrow \alpha = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{N-1} \end{pmatrix}$$

Estimator:

$$\hat{Y} = \alpha_0 + \alpha_1 X + \alpha_2 X^2 + \dots + \alpha_{N-1} X^{N-1}$$

**Simple linear regression:** Estimate line, i.e, estimate  $\alpha_0, \alpha_1$  and set  $\alpha_2 = \dots = \alpha_{N-1} = 0$

# What criterion to optimize? (->Which line is better?)

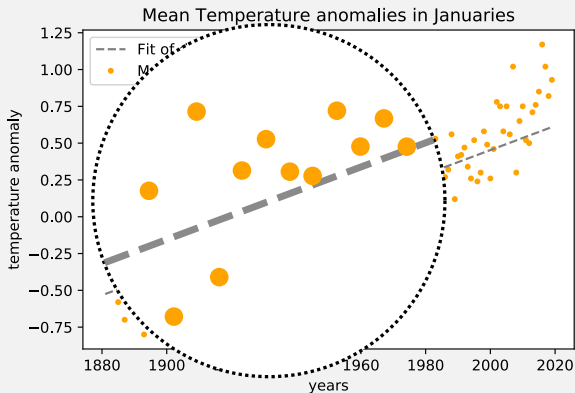


# Optimization criteria

- ❖ Residual: difference predicted/observed  $|Y_i - \hat{Y}_i|$
- ❖ Possible minimization criteria:
  - ❖ Sum of residuals
  - ❖ Maximum
  - ❖ Variance of residuals

$$\text{Var}_{res} := \frac{1}{N} \sum_i (Y_i - \hat{Y}_i)^2 = E[(Y - \hat{Y})^2]$$

- ❖ Minimize  $\text{Var}_{res}$  = ordinary least squares optimization

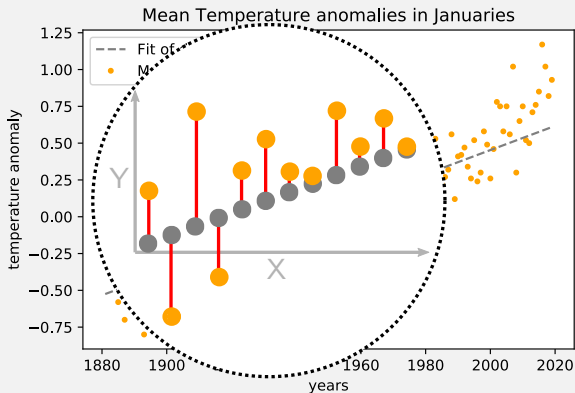


# Optimization criteria

- ❖ Residual: difference predicted/observed  $|Y_i - \hat{Y}_i|$
- ❖ Possible minimization criteria:
  - ❖ Sum of residuals
  - ❖ Maximum
  - ❖ Variance of residuals

$$\text{Var}_{res} := \frac{1}{N} \sum_i (Y_i - \hat{Y}_i)^2 = E[(Y - \hat{Y})^2]$$

- ❖ Minimize  $\text{Var}_{res}$  = ordinary least squares optimization



# *Why least squares?*

Three advantages:

1. Penalizes large deviations from the observed data very strongly and sums over all data points;
2. Finding the polynomial that minimizes the variance can be done efficiently via least squares optimization methods;
3. Minimizing the variance guarantees that there is one and only one solution.

## Coefficient of determination $R^2$

How to measure quality of fit?

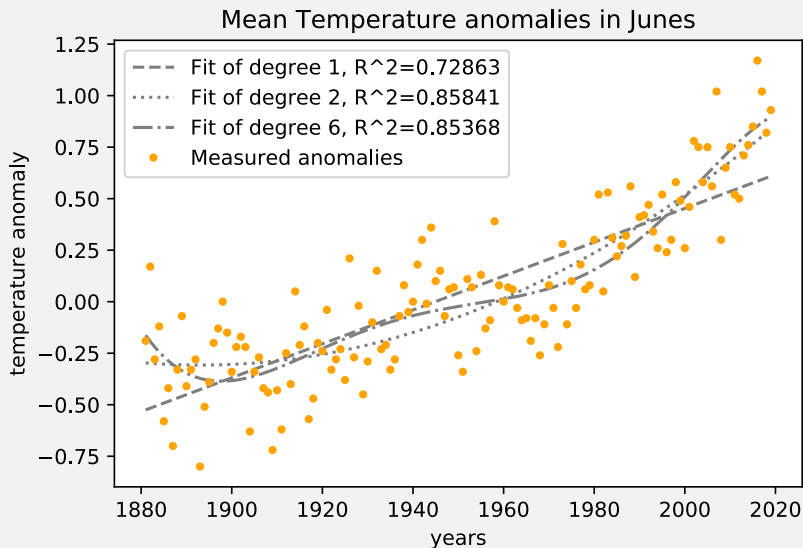
Recall: Ordinary Least squares optimization minimizes  $Var_{res}$

$R^2$  is a normalized measure thereof:

$$R^2 := 1 - \frac{Var_{res}}{Var(Y)}$$

i.e.,  $R^2 \rightarrow 0$  bad fit,  $R^2 \rightarrow 1$  good fit

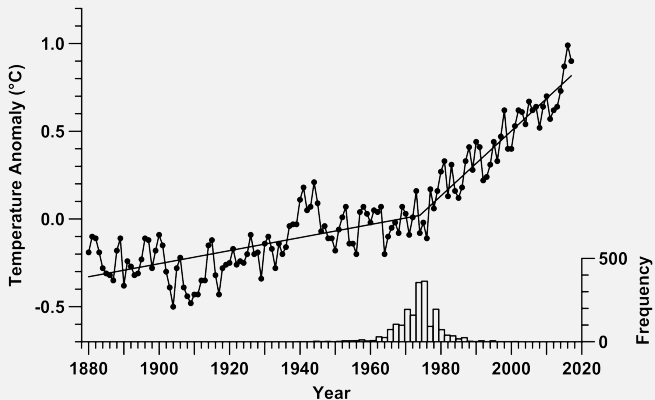
# Separated training from testing





# The “break” model

- ❖ Combination of two linear functions
- ❖ Estimated break around year  $1974 \pm 5.9$



source: Mudelsee, M. (2019). Trend analysis of climate time series: A review of methods. *Earth Science Reviews*, 190, 310–322.

# Quiz

## True or false?

- ❖ The residual is the distance between an observed and its predicted data point
  - ❖ Linear regression always minimizes the variance of residuals
  - ❖ Linear regression is the task of fitting a line to a set of data points
  - ❖ Ordinary least squares always minimizes the variance of residuals
- ❖ How does linear regression measure the distance between an observed and its predicted data point?



(a)



(b)



(c)

# Quiz

## True or false?

- ❖ The residual is the distance between an observed and its predicted data point true
  - ❖ Linear regression always minimizes the variance of residuals false
  - ❖ Linear regression is the task of fitting a line to a set of data points false
  - ❖ Ordinary least squares always minimizes the variance of residuals true
- ❖ How does linear regression measure the distance between an observed and its predicted data point? (a)



(a)



(b)



(c)

# *Recap*

# Summary

- ❖ Plots with `matplotlib`:
  - ❖ Line- and scatter plot
  - ❖ Histogram
  - ❖ Whisker (box) plot
- ❖ Numpy
  - ❖ `ndarray` data type
  - ❖ Vectorized operations, broadcasting
  - ❖ Curve fitting: `polyfit()`
- ❖ Realistic data analysis: climate trends

## What comes next?

- ❖ Draw your first plots with `matplotlib`
- ❖ Further reading about NumPy: Chapter 2 of the “Python Data Science Handbook”:  
<https://jakevdp.github.io/PythonDataScienceHandbook/>
- ❖ Due date for this week’s exercises is **Wednesday, December 20, 2pm, 2023.**

*Next lecture: Pandas, applied machine learning, etc. ...*