

Lab 0

Introduction to Python

*Data Mining for Computer and
Systems Sciences (DAMI)*




Luis Quintero



luis-eduardo@dsv.su.se



2023-08-30

DataScienceGroup
 datascience.dsv.su.se



Stockholm
University

ILOs

- I. Assess the **quality of a structured dataset**
- II. Formulate **analytical questions** that can be solved with descriptive and exploratory methods
- III. Recognize the types of questions that can be answered with **predictive methods**
- IV. Configure **Python** to design and implement a data science project

1. Structured high-quality datasets

2. Data Mining and Predictive Analytics

3. Python

DataScienceGroup
 datascience.dsv.su.se



Stockholm
University

Study case I

Bank Marketing

Bank-v1.csv

From: [Bank Marketing - UCI Machine Learning Repository](#)

	A	B	C	D	E	F	G	H	I	J	K	L
1	30	unemployed	married	primary	no	1787	no	no				
2	33	services	married	secondary	no	4789 euros	yes	yes	this person is the cousin of the owner			
3		management	single	tertiary	no	1350	yes	no				
4	30	management	married	tertiary	yes	1476	yes	yes		the person still owes money		
5	59	blue-collar	married	secondary	yes	0	yes	no				
6	35	management	single	tertiary		747	no	no				
7												
8	36	self-employed	married	tertiary		307	yes	no				
9	39	technician	married	secondary		147	yes	no				
10	41	entrepreneur	married	tertiary	no	221	yes	no				
11	43	services	married	primary	no	-88	yes	yes				
12	39	services		secondary	no	9374	yes	no				
13	43	admin.	married	secondary	no	264	yes	no				
14		technician		tertiary	no	1109	no	no				
15	20	student	single	secondary	no	502	no	no				
16	31	blue-collar	married	secondary	no	"€360"	yes	yes				
17	40	management	married	tertiary	yes	194	no	yes				
18	56	technician	married	secondary	yes	4073	no	no				
19	37	admin.	single	tertiary	no	2317	yes	no				
20	25	blue-collar	single	primary	no	-221	yes	no				
21	25	blue-collar	single	primary	no	-221	yes	no				

Saturday, February 1, 2014

Financial Report

INFORMATION

Printed:11:51:34AM

Printed By: A

Start Time:12:00 AM

End Time:11:59 PM

Employee Specific:

Date	Name	QTY	Username	TAX1 (25%)	TAX2 (25%)	TAX3 (5%)	Total
Fixed time							
2/1/2014 12:	60 minutes	0.17	80 minkoon	-	-	-	10.00
2/1/2014 12:	120 minutes	0.17	120 minkoon	-	-	-	20.00
2/1/2014 12:	60 minutes	0.17	80 MPR MPRM	-	-	-	10.00
2/1/2014 1:	60 minutes	0.17	80 backstage	-	-	-	10.00
2/1/2014 2:	120 minutes	0.08	120 mri21	-	-	-	10.00
2/1/2014 2:	180 minutes	0.17	180 minkoon	-	-	-	30.00
2/1/2014 3:	120 minutes	0.17	120 vantage	-	-	-	20.00
2/1/2014 4:	60 minutes	0.08	80 mri21	-	-	-	5.00
2/1/2014 4:	60 minutes	0.08	80 mri21	-	-	-	5.00
2/1/2014 4:	60 minutes	0.08	80 mri21	-	-	-	5.00
2/1/2014 8:	120 minutes	0.17	120 gamet187	-	-	-	20.00
2/1/2014 11:	180 minutes	0.17	180 minkoon	-	-	-	30.00
Sub group total				0.00	0.00	0.00	175.00
Group Total				0.00	0.00	0.00	175.00

Task: Identify possible problems in the previous dataset

For example:

- Lack of **column names** with descriptions
- Some cells have **missing values**
- The **format** in the same column is inconsistent
- **Empty rows**
- **Duplicates** in the last two rows
- **Unstructured data** in random cells (imgs)

Bank-v2.csv

File Home Insert Page Layout Formulas Data Review View Help Tell me what you want to do								
Clipboard		Font		Alignment		Number		Styles
Cut Copy Paste Format Painter		Calibri 11 A A B I U [Color] [Text Color]		[Bullet] [Numbered] [List Group] Merge & Center		General [Format] % .00 [Increase] [Decrease]		Normal Bad Good Check Cell Explanatory ... Input
L10								
	A	B	C	D	E	F	G	H
1	age	job	marital	education	default	balance	housing	loan
2	30	unemployed	married	primary	no	1787	no	no
3	33	services	married	secondary	no	4789	yes	yes
4	35	management	single	tertiary	no	1350	yes	no
5	30	management	married	tertiary	no	1476	yes	yes
6	59	blue-collar	married	secondary	no	0	yes	no
7	35	management	single	tertiary	no	747	no	no
8	36	self-employed	married	tertiary	no	307	yes	no
9	39	technician	married	secondary	no	147	yes	no
10	41	entrepreneur	married	tertiary	no	221	yes	no
11	43	services	married	primary	no	-88	yes	yes
12	39	services	married	secondary	no	9374	yes	no
13	43	admin.	married	secondary	no	264	yes	no
14	36	technician	married	tertiary	no	1109	no	no
15	20	student	single	secondary	no	502	no	no
16	31	blue-collar	married	secondary	no	360	yes	yes
17	40	management	married	tertiary	no	194	no	yes
18	56	technician	married	secondary	no	4073	no	no
19	37	admin.	single	tertiary	no	2317	yes	no
20	25	blue-collar	single	primary	no	-221	yes	no

Structured dataset

Row, observation,
sample, registry,
item, subject

Columns, variables,
attributes, features

age	job	marital	education	default	balance	housing	loan
30	unemployed	married	primary	no	1787	no	no
33	services	married	secondary	no	4789	yes	yes
35	management	single	tertiary	no	1350	yes	no
30	management	married	tertiary	no	1476	yes	yes
59	blue-collar	married	secondary	no	0	yes	no
35	management	single	tertiary	no	747	no	no
36	self-employed	married	tertiary	no	307	yes	no

Cell, field, value

Bank-v2-metadata.csv

From Wikipedia: **Metadata** (or **metainformation**) is "data that provides information about other data",^[1] but not the content of the data, such as the text of a message or the image itself.^[2]

#	Column name	Description	Data Type	Details
1	age	age	numeric	
2	job	type of job	categorical	"admin.", "unknown", "unemployed", "management", "housemaid", "entrepreneur", "student", "blue-collar", "self-employed", "retired", "technician", "services"
3	marital	marital status	categorical	"married", "divorced", "single"; note: "divorced" means divorced or widowed
4	education	education level	categorical	"unknown", "secondary", "primary", "tertiary"
5	default	has credit in default?	binary	"yes", "no"
6	balance	average yearly balance	numeric	in euros
7	housing	has housing loan?	binary	"yes", "no"
9 8	loan	has personal loan?	binary	"yes", "no"

Types of variables / Data types

Numerical

- **Discrete** – *Only certain values are possible*
 - User ID, Age, # children, Year, Frequencies, ...
- **Continuous** – *Any value within a range*
 - Height, balance \$, T°, joystick values, time [ms, s]...

Categorical

- **Nominal** – *Labels without quantitative meaning*
 - Non-binary: Gender, Country, Role, Marital status, Team, ...
 - **Binary**: Is Smoker? Is Active? Is Subscribed?
- **Ordinal** – *Labels with relative order/rank*
 - Education Level, Age Group, Likert Scales, any other scale...
 - **Binary**: Bad/Good? Low/High?, ...

Which data type is a variable **PhoneNumber**? 071 123 4567
The fact that it contains numbers **does not mean** it is numerical!

Data quality

“If it cannot be guaranteed that the data is completely clean, the subsequent analysis lacks any scientific rigor and its ability to be useful for the purpose we seek.”

Xavi Font - Business Intelligence y Business Analytics (2019)

6 dimensions of data quality

Accuracy

Do the data reflect reality?

E.g., age containing negative numbers

Timeliness

Are the data available when needed?

E.g., the end-of-the-month report has access to all the required metrics

Validity

Does the format follow the specific business rules?

E.g., define a date as "23/08/2023" or "23-08-23"

Completeness

Are the data complete?

E.g., the metadata describing the data is missing, or definition of the optional fields

Consistency

Do the data from different sources follow the same structure?

E.g., databases from different campuses at SU

Uniqueness

Does every instance appear only once in the dataset?

E.g., perhaps the person John William Smith and John W. Smith are the same subject.

Task: Think of 2 questions that can be answered from the structured dataframe and the process to answer them

E.g., What is the average balance of married people?
(*filter marital, then mean on balance*)

	A	B	C	D	E	F	G	H
1	age	job	marital	education	default	balance	housing	loan
2	30	unemployed	married	primary	no	1787	no	no
3	33	services	married	secondary	no	4789	yes	yes
4	35	management	single	tertiary	no	1350	yes	no
5	30	management	married	tertiary	no	1476	yes	yes
6	59	blue-collar	married	secondary	no	0	yes	no
7	35	management	single	tertiary	no	747	no	no
8	36	self-employed	married	tertiary	no	307	yes	no
9	39	technician	married	secondary	no	147	yes	no
10	41	entrepreneur	married	tertiary	no	221	yes	no
11	43	services	married	primary	no	-88	yes	yes
12	39	services	married	secondary	no	9374	yes	no
13	43	admin.	married	secondary	no	264	yes	no
14	36	technician	married	tertiary	no	1109	no	no
15	20	student	single	secondary	no	502	no	no
16	31	blue-collar	married	secondary	no	360	yes	yes
17	40	management	married	tertiary	no	194	no	yes
18	56	technician	married	secondary	no	4073	no	no
19	37	admin.	single	tertiary	no	2317	yes	no
20	25	blue-collar	single	primary	no	-221	yes	no

	Colname	Description	Data Type
1	age	age	numeric
2	job	type of job	categorical
3	marital	marital status	categorical
4	education	education level	categorical
5	default	has credit in default?	binary
6	balance	average yearly balance	numeric
7	housing	has housing loan?	binary
8	loan	has personal loan?	binary

Univariate (1 variable or column)

- **How many managers** are in the customers list? *(filter and count on the column `job`)*
- What is the **range** of the **balance** in the bank accounts of the customers? *(min and max on the column `balance`)*
- What is the **proportion** of customers with personal **loan**? *(count customers where `loan` = "yes" / # customers)*

Bivariate (2 variables or columns)

- What is the **proportion** of **students** that are **married**? *(calculation on the columns `job` and `marital status`)*

Multivariate (>2 variables or columns)

- **Which students** under **25** y.o. have more than **1000** euros? *(job, age, balance)*
- What is the **average balance** of **married** people with minimum **secondary** education? *(pivot table on the involved columns and aggregation function)*

Formulating analytical questions

Every analytical question leads to a certain **analytical approach** and specific involved **variables/features/columns**.

The value of the data can be discovered asking the right questions!

More **specific** questions lead to more valuable and processable answers.



Which questions can be answered from this dashboard?

Which questions can be answered from this dashboard?



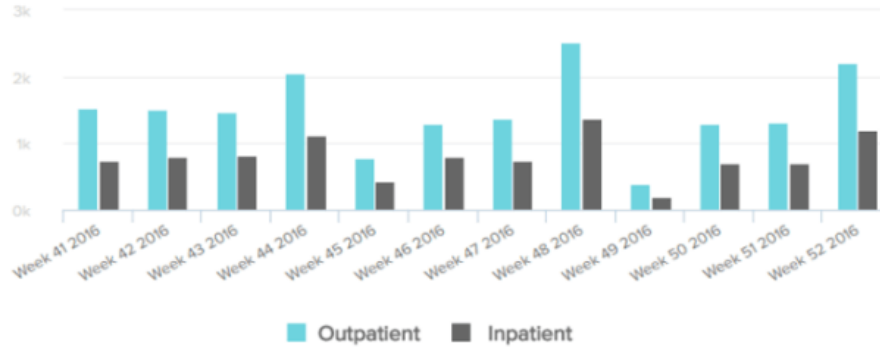
50,001
Total Patients

34,863
Total Admissions

\$ 8,742
Avg Treatment Costs

53min
Avg ER Wait Time

Outpatients vs. Inpatients Trend

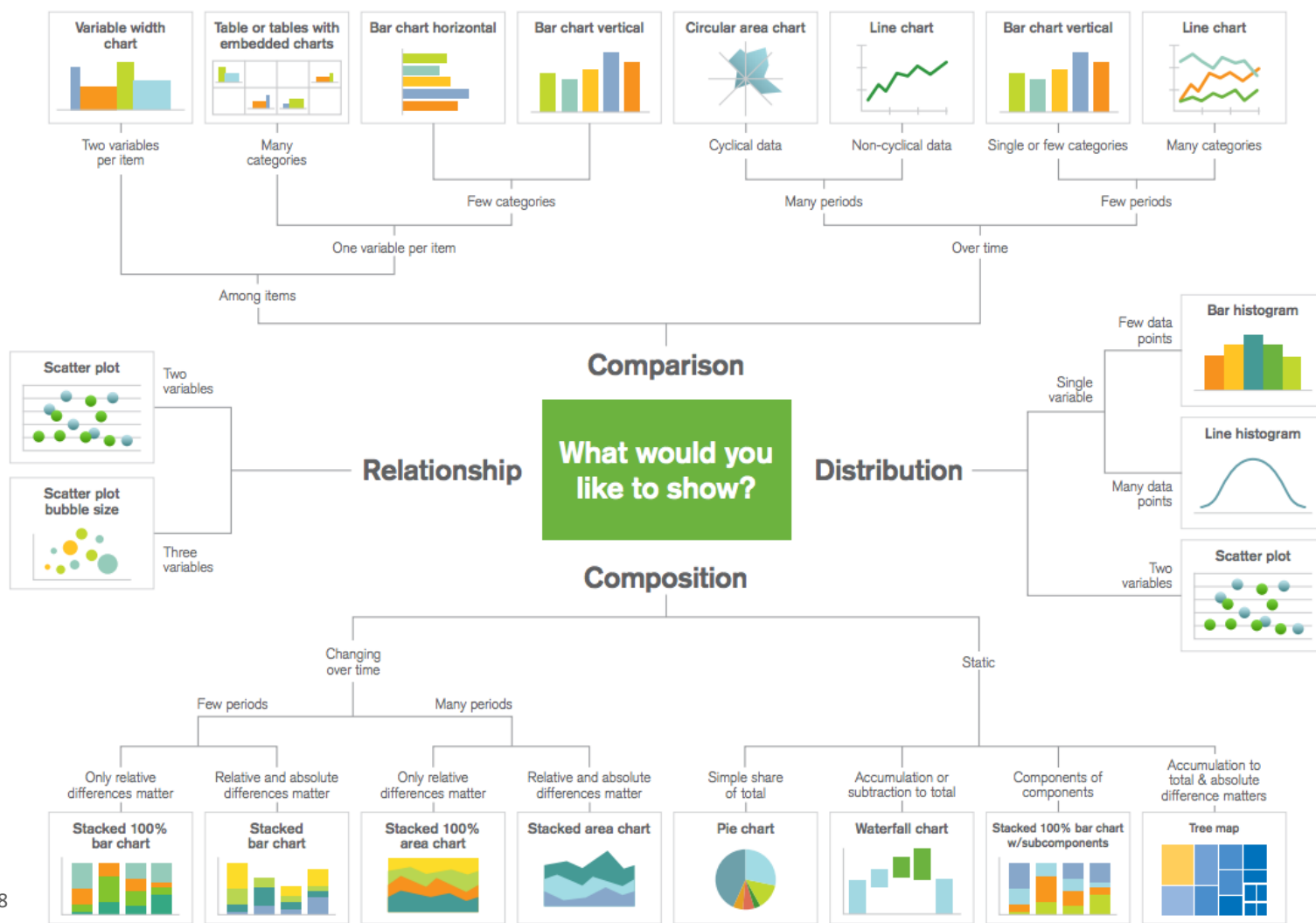


Patients By Division

division	patient_status	
	inpatient	outpatient
Surgery	9.471 ↑	17.642
Gynaecology	6.869 ↑	13.053
Dermatology	5.299	9.772 ↑
Neurology	3.540	6.581
Oncology	3.088 ↓	5.842
Orthopaedics	2.809	5.144
Cardiology	2.046	3.868 →

Avg Waiting Time By Division





1. Structured high-quality datasets

2. Data Mining and Predictive Analytics

3. Python

DataScienceGroup
datascience.dsv.su.se

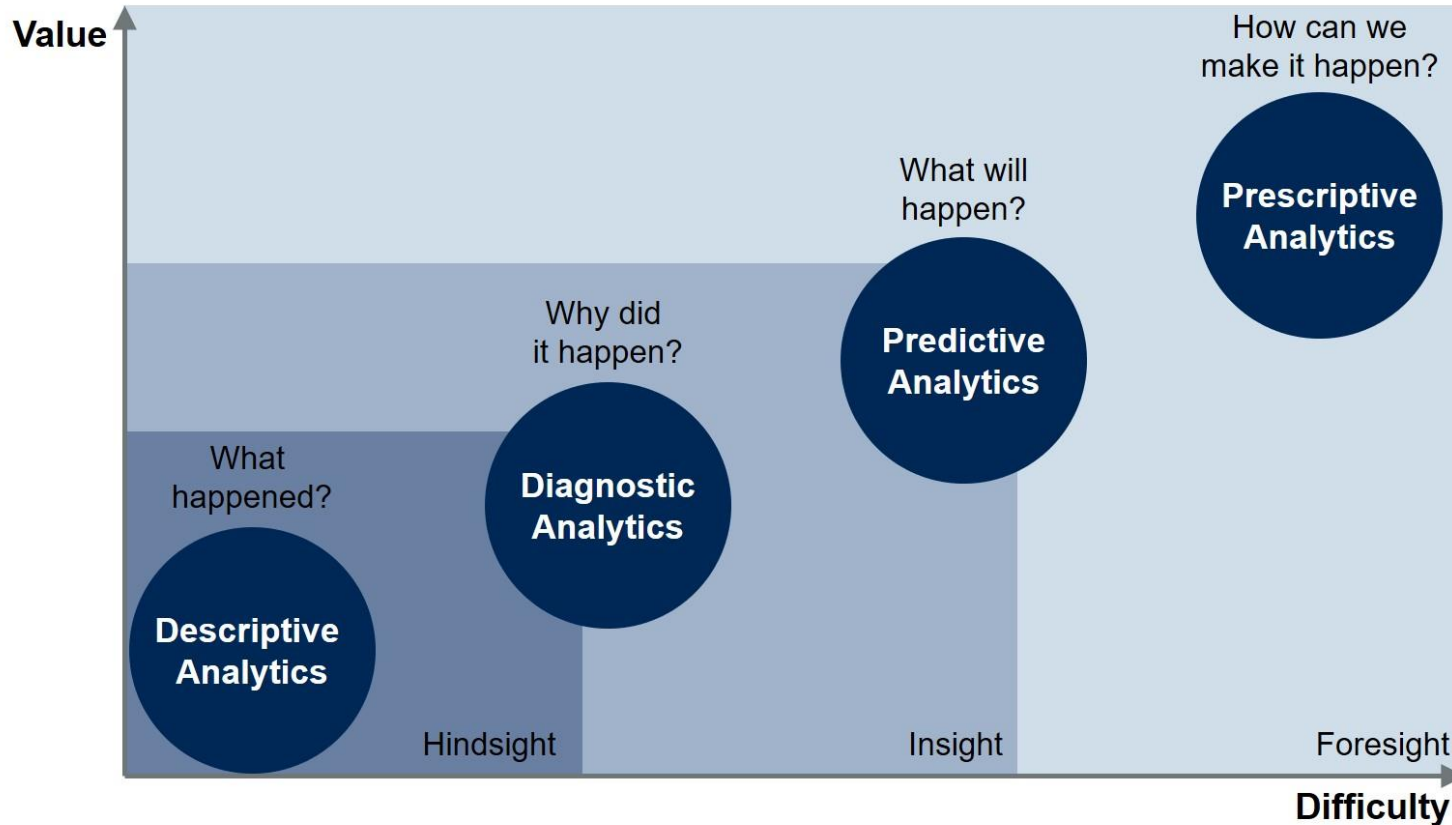


Stockholm
University

- ✓ Structured data
- ✓ Descriptive metadata
- ✓ Clean data
- ✓ Well-defined questions
- ✓ Descriptive analytical approaches

How is data mining and machine learning different from data analytics in Excel?

Four types of data analytics



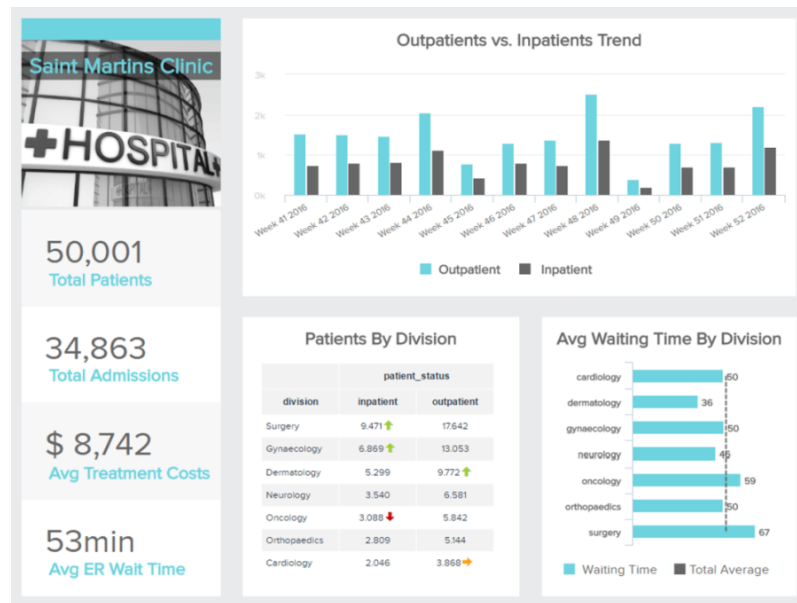
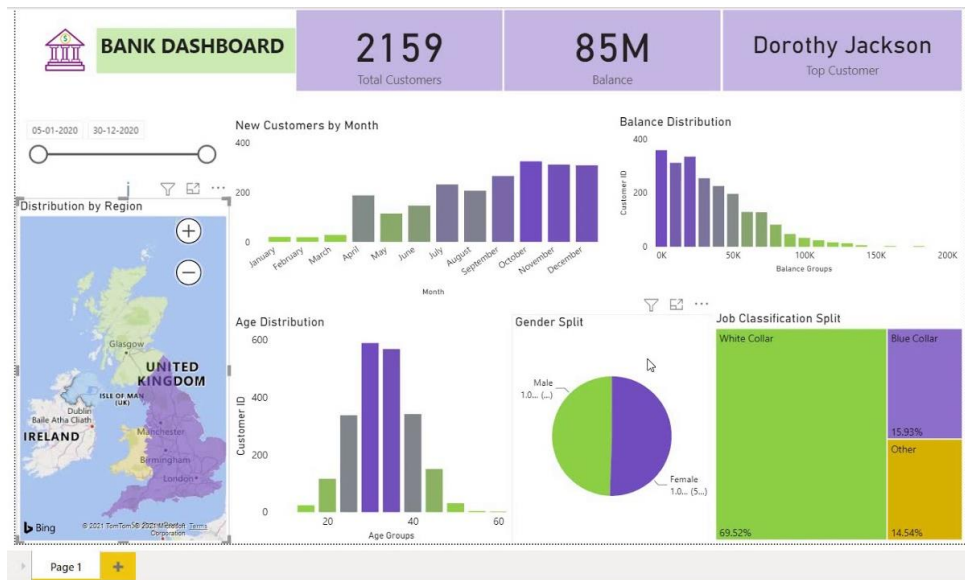
Exploratory and Descriptive Analytics

It is introductory, retrospective and answers to the question: **What happened?**

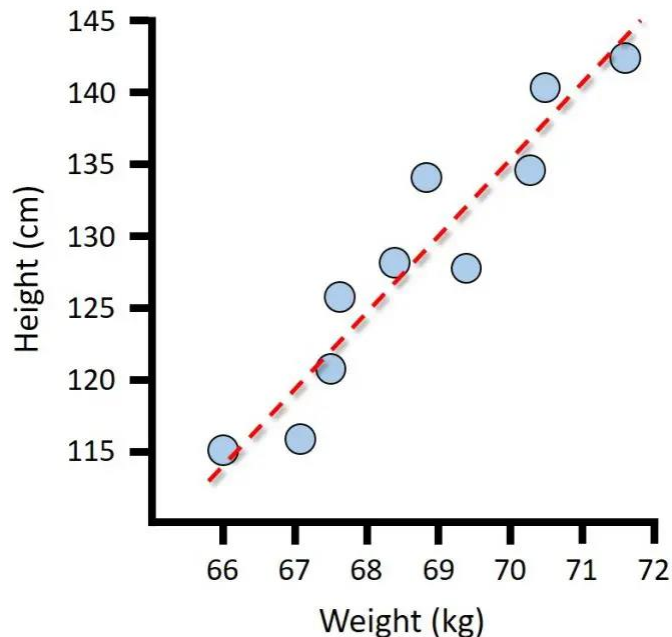
Most data analyses in a business are descriptive, which makes it the most common type.

EDAs are usually visualized in simple reports, control panels, or KPI dashboards in software visualization frameworks.

EDAs are usually enough to answer simple questions



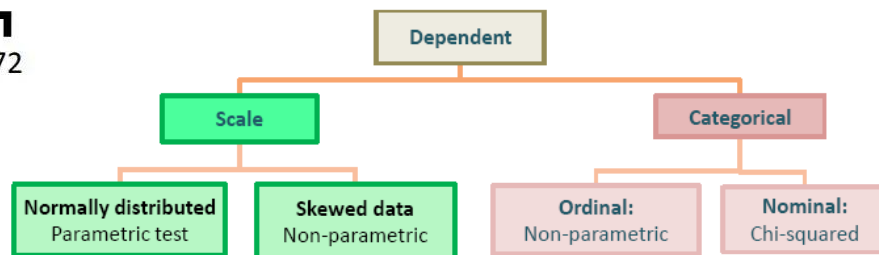
Diagnostic analytics



Answers to the question:
Why did something happened?

Some approaches:

- Correlation analysis
- Inferential statistics
- Clustering



**But you can ask more
complicated questions
from the data...**

Task: How many groups of similar customers can be created from the bank dataset?



How do you group them? Age, job, or balance,... ?



File Home Insert Page Layout Formulas Data Review View Help Tell me what you want to do																			
Paste		Clipboard		Font				Alignment				Number		Conditional Formatting		Format as Table		Styles	
Cut Copy		Format Painter		Calibri 11 A A				Wrap Text Merge & Center				%		General		Normal		Bad Good Neutral	
				B I U										Check Cell		Explanatory ...		Input Linked Cell	
J22																			
	A	B	C	D	E	F	G	H	I										
1	age	job	marital	education	default	balance	housing	loan	GROUP										
2	30	unemployed	married	primary	no	1787	no	no	1										
3	33	services	married	secondary	no	4789	yes	yes	1										
4	35	management	single	tertiary	no	1350	yes	no	2										
5	30	management	married	tertiary	no	476	yes	yes	1										
6	59	blue-collar	married	secondary		0	yes	no	3										
7	35	management	single	tertiary		47	no	no	2										
8	36	self-employed	married			7	yes	no	1										
9	39	technician	married				yes	no	3										
10	41	entrepreneur	married			221	yes	no	2										
11	43	services	married			-88	yes	yes	1										
12	39	services	married		no	9374	yes	no	2										
13	43	admin.	married	secondary	no	264	yes	no	3										
14	36	technician	married	tertiary	no	1109	no	no	2										
15	20	student	single	secondary	no	502	no	no	1										
16	31	blue-collar	married	secondary	no	360	yes	yes	2										
17	40	management	married	tertiary	no	194	no	yes	3										
18	56	technician	married	secondary	no	4073	no	no	3										
19	37	admin.	single	tertiary	no	2317	yes	no	2										
20	25	blue-collar	single	primary	no	-221	yes	no	1										

Clustering task!

Types of data analytics

DESCRIPTIVE

DIAGNOSTIC

PREDICTIVE

PRESCRIPTIVE

1 PHASES

Past data

Analysis

Results

Analyzes **what** happened in the past

- KPI
- Dashboards
- Descriptive analytics (e.g., frequencies, mean, std, ...)
- Filters
- Pivot tables

1

Past data

Build a model

Results.

Analyzes **why** something happened in the past **without** verifying with future data

- Inferential statistics (significance tests)
- Correlations
- Clustering (e.g., K-Means)

1

Past data

Build a model

Results

2

New data

Apply the model

Prediction

Predicts **what will happen** in the future based on the past and checks how **accurate** the predictions are

- Cross-validation
- Machine learning (e.g., KNN, SVM, DT, RF, DL, ...)
- Simulations

1

Past data

Build a model

Results

2

New data

Apply the model

Prediction

3

... and prescribes actions based on the predictions.

- Feature importance
- Counterfactuals
- Explainable ML

Apply the prediction

Action

DataScienceGroup
datascience.dsv.su.se



Stockholm University

Study case II

Plant classification

Non-structured datasets

Edgar Anderson went on an expedition to quantify the morphologic variations of the iris flowers of three species. The data was collected in the Gaspé Peninsula in southern Canada “all from the same pasture, picked up on the same day and measured at the same time by the same person with the same apparatus”.



Iris Versicolor



Iris Setosa



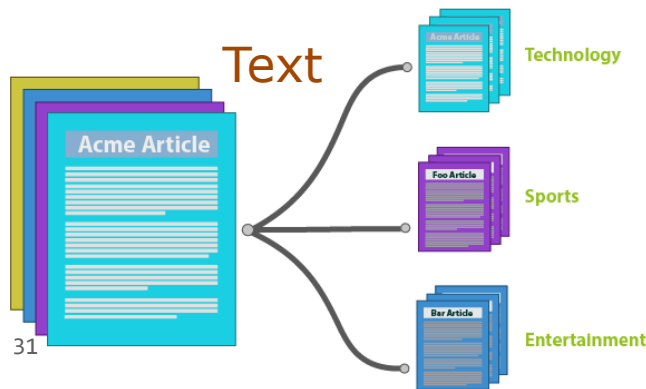
Iris Virginica

Data Modalities

	age (n)	job (c)	marital (c)	education (c)	balance (n)	housing (c)
0	30	unemployed	married	primary	1787	no
1	33	services	married	secondary	4789	yes
2	35	management	single	tertiary	1350	yes
3	30	management	married	tertiary	1476	yes
4	59	blue-collar	married	secondary	0	yes
5	35	management	single	tertiary	747	no

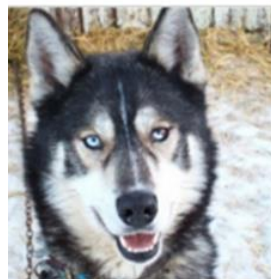
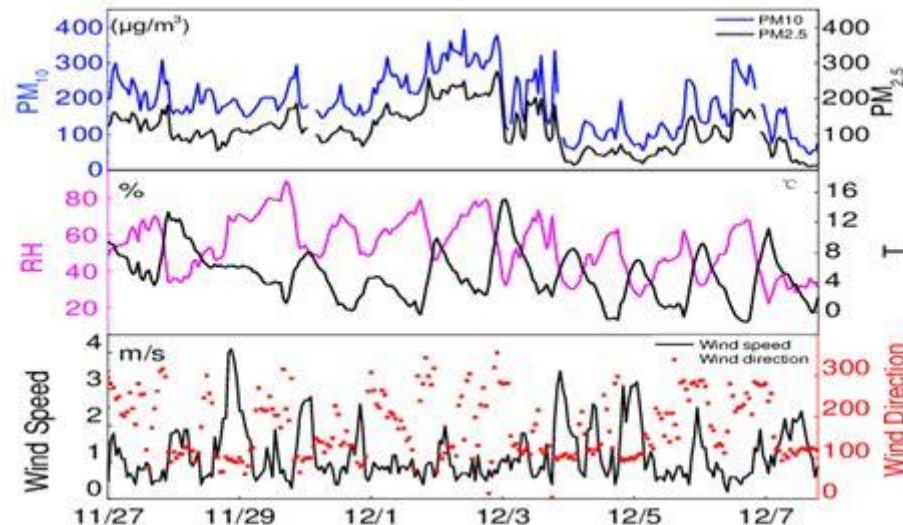
Tabular/Structured

(we will use this modality during the labs!)

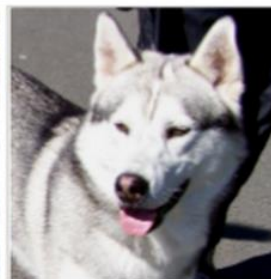


31

Time series



Predicted: **wolf**
True: **husky**



Predicted: **husky**
True: **husky**



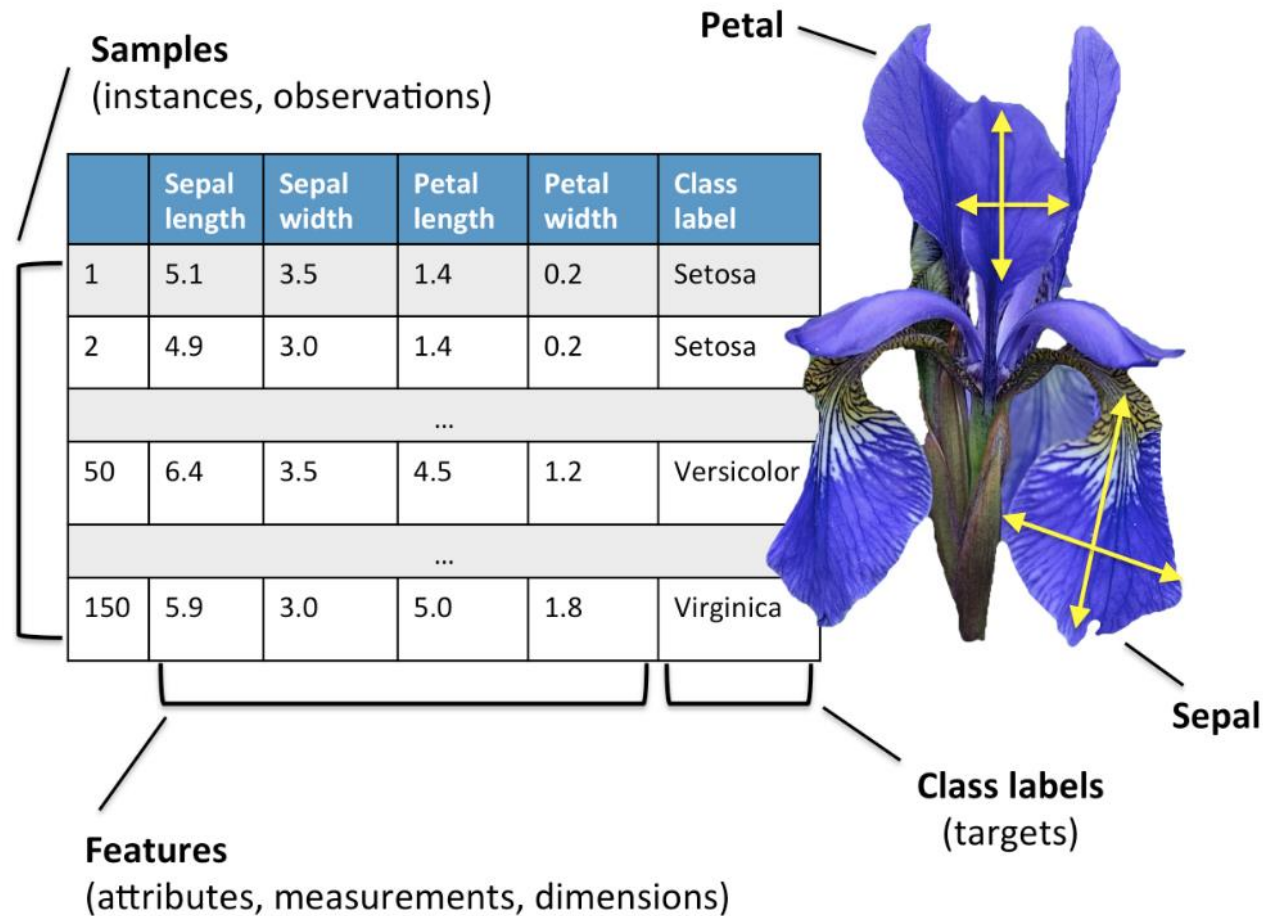
Predicted: **wolf**
True: **wolf**

Images/Video

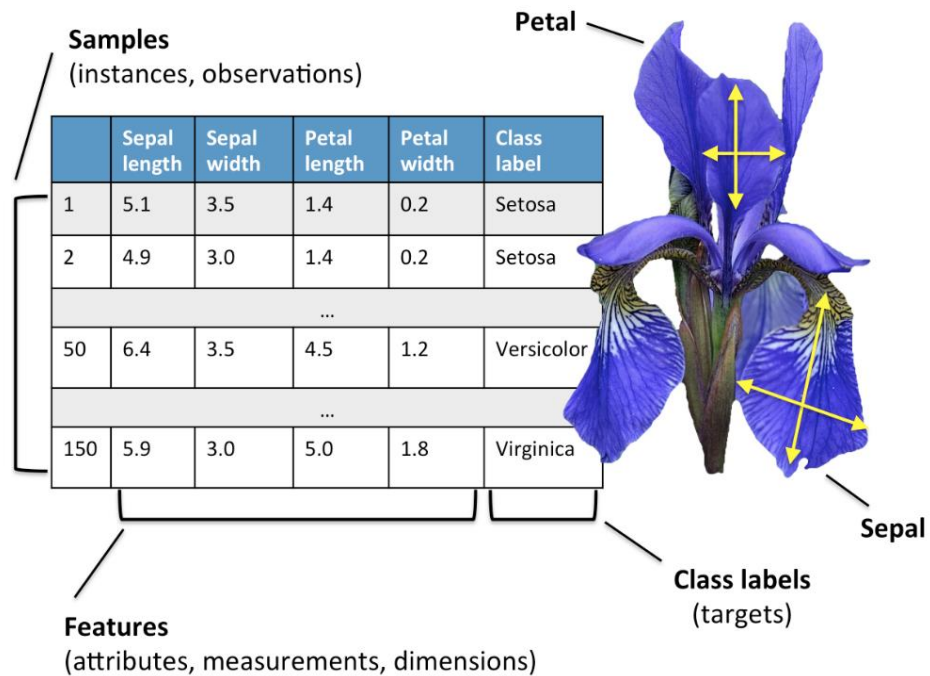
DataScienceGroup
datascience.dsv.su.se



Stockholm
University



	A	B	C	D	E
1	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa
6	5	3.6	1.4	0.2	setosa
7	5.4	3.9	1.7	0.4	setosa
8	4.6	3.4	1.4	0.3	setosa
9	5	3.4	1.5	0.2	setosa
10	4.4	2.9	1.4	0.2	setosa
11	4.9	3.1	1.5	0.1	setosa
52	7	3.2	4.7	1.4	versicolor
53	6.4	3.2	4.5	1.5	versicolor
54	6.9	3.1	4.9	1.5	versicolor
55	5.5	2.3	4	1.3	versicolor
56	6.5	2.8	4.6	1.5	versicolor
57	5.7	2.8	4.5	1.3	versicolor
58	6.3	3.3	4.7	1.6	versicolor
59	4.9	2.4	3.3	1	versicolor
60	6.6	2.9	4.6	1.3	versicolor
61	5.2	2.7	3.9	1.4	versicolor
102	6.3	3.3	6	2.5	virginica
103	5.8	2.7	5.1	1.9	virginica
104	7.1	3	5.9	2.1	virginica
105	6.3	2.9	5.6	1.8	virginica
106	6.5	3	5.8	2.2	virginica
107	7.6	3	6.6	2.1	virginica
108	4.9	2.5	4.5	1.7	virginica
109	7.3	2.9	6.3	1.8	virginica
110	6.7	2.5	5.8	1.8	virginica
111	7.2	3.6	6.1	2.5	virginica



Task: Which questions can be asked from this dataset?

Predictive Task

Based on the 150 samples collected previously in the iris dataset.

Create a model that allows us to determine whether a **new** iris flower is either:

- setosa
- virginica
- versicolor

	A	B	C	D	E
1	SepalLengthCm ▾	SepalWidthCm ▾	PetalLengthCm ▾	PetalWidthCm ▾	Species ▾
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa
6	5	3.6	1.4	0.2	setosa
52	7	3.2	4.7	1.4	versicolor
53	6.4	3.2	4.5	1.5	versicolor
54	6.9	3.1	4.9	1.5	versicolor
55	5.5	2.3	4	1.3	versicolor
56	6.5	2.8	4.6	1.5	versicolor
102	6.3	3.3	6	2.5	virginica
103	5.8	2.7	5.1	1.9	virginica
104	7.1	3	5.9	2.1	virginica
105	6.3	2.9	5.6	1.8	virginica
151	5.9	3	5.1	1.8	virginica
152	5.2	3.1	3.5	1.4	?
153	6.5	3.5	1.6	1.3	?
154	5	3	4	0.5	?

**new
samples!!**

Iris Data (red=setosa,green=versicolor,blue=virginica)

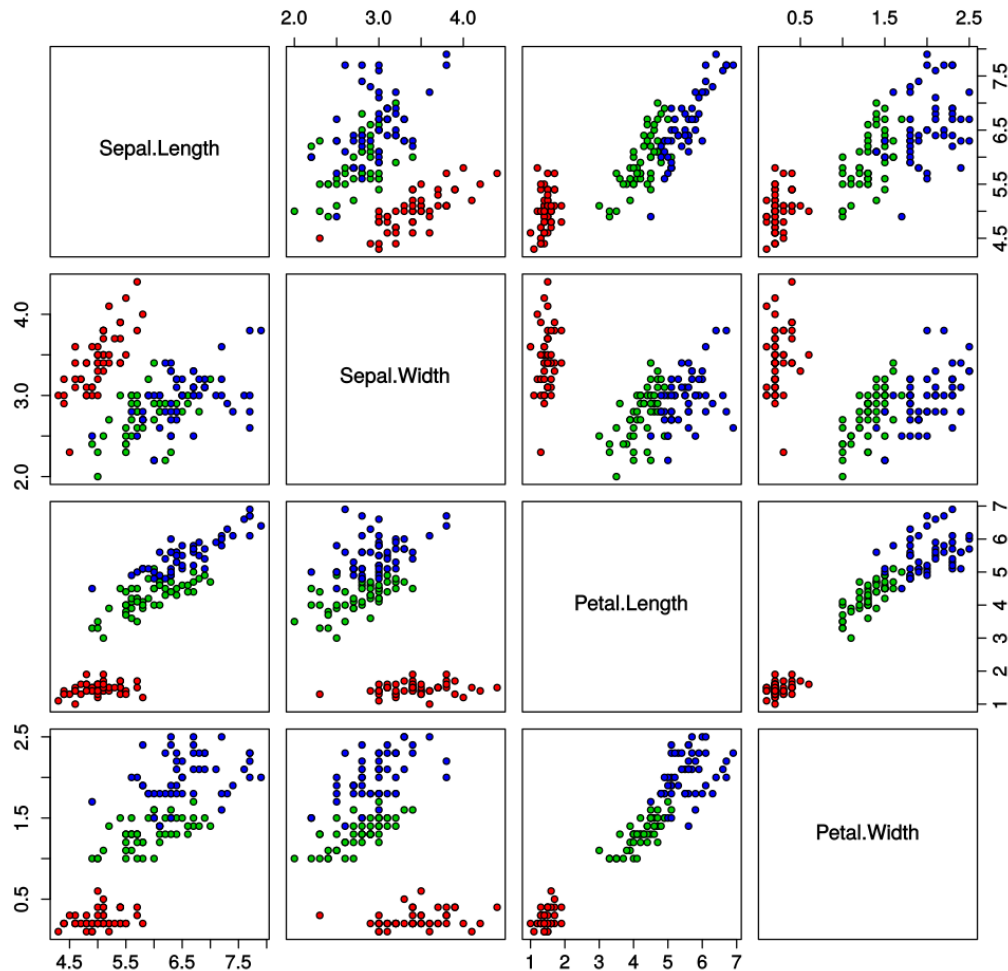


Image from: [Iris dataset scatterplot - Iris flower data set - Wikipedia](#)

A linear model

The dataset is so famous because it easily unveils predictive variables visually.

A pair plot let us see which pair of variables are more likely to separate the three species **accurately** with a **linear model**.

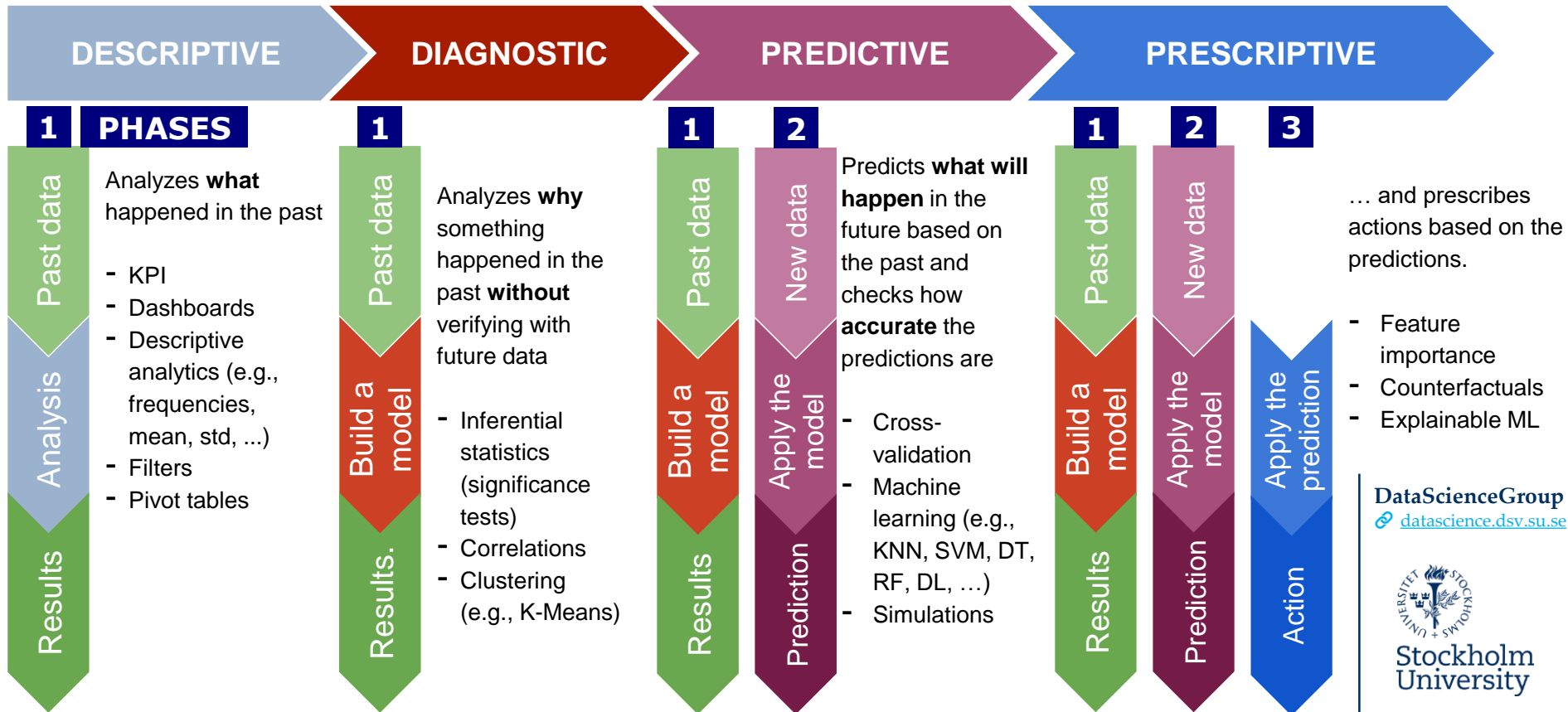
Predictive analytics

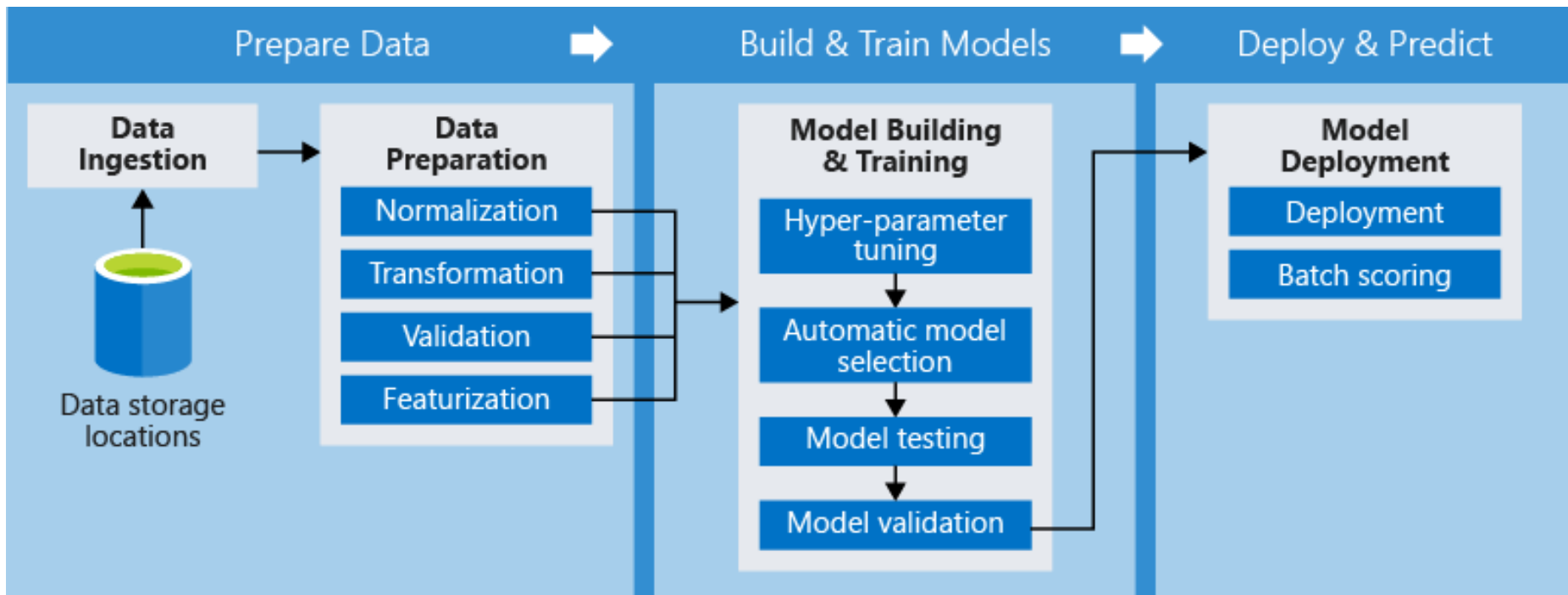
Generally applies statistical algorithms and **machine learning** techniques to answer questions related to **future values based on historical data**.

E.g.,

- K-nearest Neighbors (KNN)
- Decision Trees (DT)
- Random Forest (RF)
- Support Vector Machines (SVM)
- Deep Learning (DL)

Types of data analytics





We will work with preprocessed datasets to understand how to prepare the data, build and fine tune models, and deploy them on applications.

During the rest of the labs

Understanding the **practicalities** of new analytical approaches that answer specific type of questions:

- Which are the frequent patterns and associations from a large dataset? (**rule mining**)
- Which groups of observations are similar to each other? How many groups? (**clustering**)
- What would be the value of a specific column in a new observation based on prior experience in the same problem? (**classification**)

1. Structured high-quality datasets
2. Data Mining and Predictive Analytics

3. Python

DataScienceGroup
 datascience.dsv.su.se



Stockholm
University

All the files for the Lab session should be downloaded once.

Lab 0: How to start programming in Python?

- Lab 1:** How to understand and process my original dataset to create a machine learning model?
- Lab 2-3:** Which type of models can I train based on the characteristics of my data?
- Lab 4:** Having multiple models, how can I evaluate which one might perform better on unseen data?
- Lab 5:** How can I reuse my trained model in a production-ready environment?

DAMI Analytics - Lab Example

This app classifies three varieties of wheat seeds from seven real-value attributes extracted from soft X-rays

More information about dataset <https://archive.ics.uci.edu/ml/datasets/seeds>

Classification with Trained Model



Press below to execute the classification

EXECUTE CLASSIFICATION

Possible classes: [0:Kama], [1:Rosa], [2:Canadian]

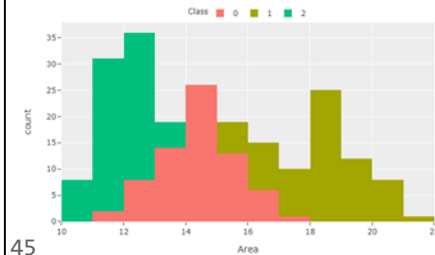
Dataset Visualization

The next plots show some characteristics of the original dataset. Note that the values from the SAMPLE that was input above will be highlighted in the plot according to the selected variables.

Histogram per class of a variable

Choose a variable:

Area

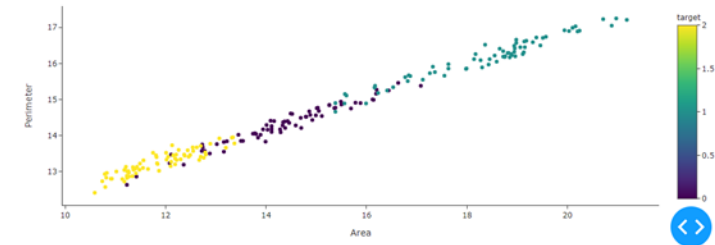


Scatter plot of two variables

Choose two variables to plot:

Area

Perimeter



The interactive web platform allows manipulation of input variables and data visualization of the input compared to original dataset.

DataScienceGroup
datascience.dsv.su.se



Tools

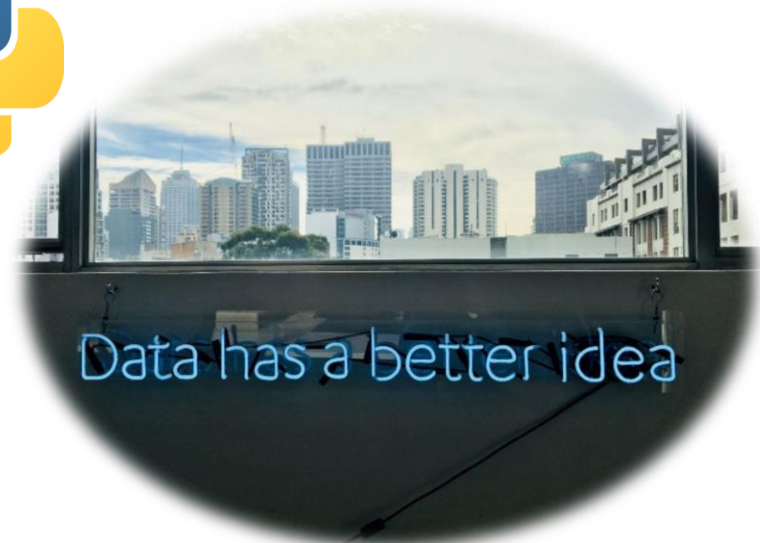
Python programming language



Why not Excel?

It has limitations regarding:

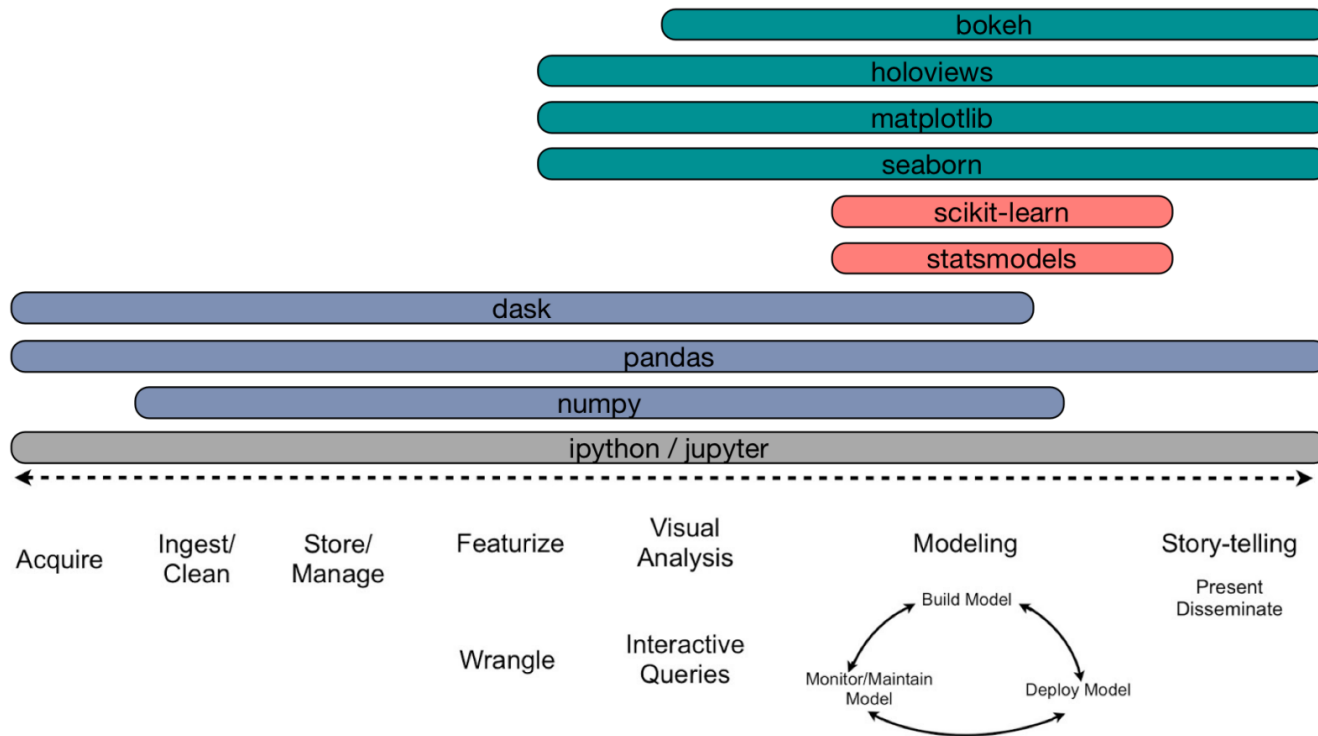
- Manipulation of large datasets
- Reproducibility of analysis pipelines
- Not compatible to train machine learning models



DataScienceGroup
datascience.dsv.su.se



Python Ecosystem for Data Science



Documentation

Python » English » 3.8.3 » Documentation » The Python Tutorial »

Table of Contents

- 5. Data Structures
 - 5.1. More on Lists
 - 5.1.1. Using Lists as Stacks
 - 5.1.2. Using Lists as Queues
 - 5.1.3. List Comprehensions
 - 5.1.4. Nested List Comprehensions
 - 5.2. The `del` statement
 - 5.3. Tuples and Sequences
 - 5.4. Sets
 - 5.5. Dictionaries
 - 5.6. Looping Techniques
 - 5.7. More on Conditions
 - 5.8. Comparing Sequences and Other Types

Previous topic

4. More Control Flow Tools

Next topic

6. Modules

This Page

Report a Bug
Show Source

5. Data Structures

This chapter describes some things you've learned about already in more detail, and adds some new things as well.

5.1. More on Lists

The list data type has some more methods. Here are all of the methods of list objects:

list.append(x)

Add an item to the end of the list. Equivalent to `a[len(a):] = [x]`.

list.extend(iterable)

Extend the list by appending all the items from the iterable. Equivalent to `a[len(a):] = iterable`.

list.insert(i, x)

Insert an item at a given position. The first argument is the index of the element before which to insert, so `a.insert(0, x)` inserts at the front of the list, and `a.insert(len(a), x)` is equivalent to `a.append(x)`.

list.remove(x)

Remove the first item from the list whose value is equal to `x`. It raises a `ValueError` if there is no such item.

list.pop([i])

Remove the item at the given position in the list, and return it. If no index is specified, `a.pop()` removes and returns the last item in the list. (The square brackets around the `i` in the method signature denote that the parameter is optional, not that you should type square brackets at that position. You will see this notation frequently in the Python Library Reference.)

list.clear()

Remove all items from the list. Equivalent to `del a[:]`.

list.index(x[, start[, end]])

Return zero-based index in the list of the first item whose value is equal to `x`. Raises a `ValueError` if there is no such item.

- Usually we program having at arm's distance the documentation of the packages we plan to use.

Python: <https://docs.python.org/3/contents.html>
Numpy: <https://numpy.org/doc/>
Pandas: <https://pandas.pydata.org/docs/reference/>
Sk-learn: <https://scikit-learn.org/>
Matplotlib: <https://matplotlib.org/stable/api/>
Seaborn: <https://seaborn.pydata.org/examples/>

DataScienceGroup
datascience.dsv.su.se

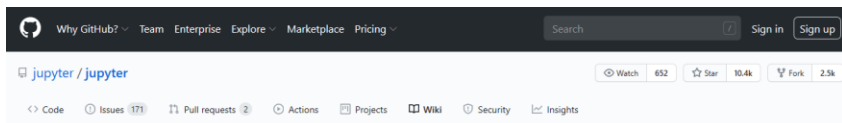


Stockholm
University

Relevant Python Packages/Modules

- Python packages/modules
 - Numpy (*vectorized operations*)
 - Pandas (*dataframe operations*)
 - Scikit-learn (*machine learning*)
 - Matplotlib (*data visualization*)
 - Seaborn (*easier data visualization*)
- Jupyter notebooks
 - Provide interactive workflow for DS
 - Combines text, math, and code in a single document

Example of Jupyter notebooks



A gallery of interesting Jupyter Notebooks

Dima Goldenberg edited this page 21 days ago · 119 revisions

This page is a curated collection of Jupyter/iPython notebooks that are notable. Feel free to add new content here, but please try to only include links to notebooks that include interesting visual or technical content; this should not simply be a dump of a Google search on every ipynb file out there.

Important contribution instructions: If you add new content, please ensure that for any notebook you link to, the link is to the rendered version using [nbviewer](#), rather than the raw file. Simply paste the notebook URL in the nbviewer box and copy the resulting URL of the rendered version. This will make it much easier for visitors to be able to immediately access the new content.

Note that [Matt Davis](#) has conveniently written a set of [bookmarklets and extensions](#) to make it a one-click affair to load a Notebook URL into your browser of choice, directly opening into nbviewer.

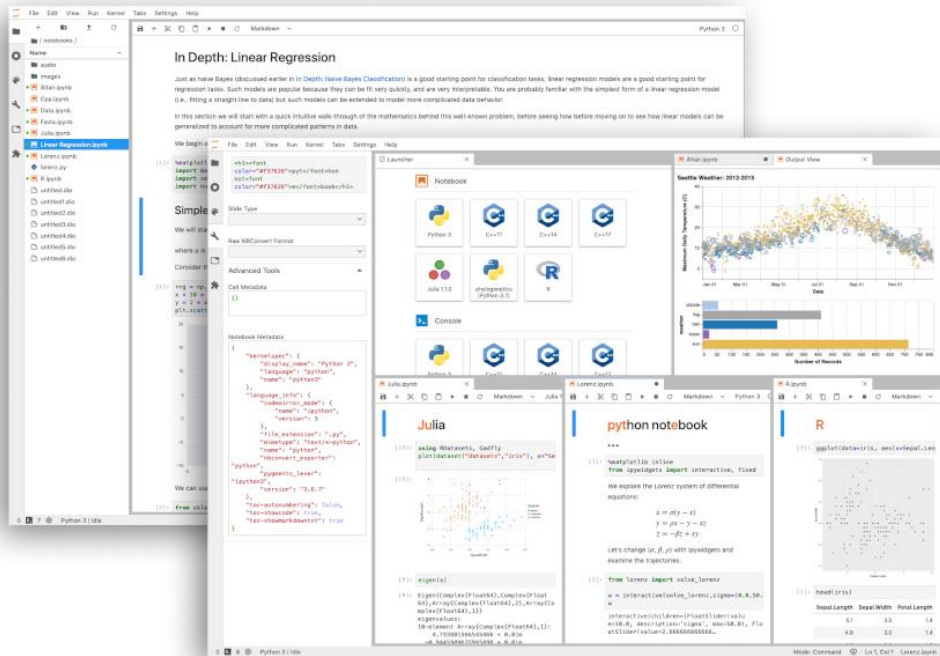
Table of Contents

1. Entire books or other large collections of notebooks on a topic
 - Introductory Tutorials
 - Programming and Computer Science
 - Statistics, Machine Learning and Data Science
 - Mathematics, Physics, Chemistry, Biology
 - Earth Science and Geo-Spatial data
 - Linguistics and Text Mining
 - Signal Processing
 - Engineering Education
2. Scientific computing and data analysis with the SciPy Stack
 - General topics in scientific computing
 - Social data

Pages 11
Find a Page...
Home
A gallery of interesting Jupyter and iPython Notebooks
A gallery of interesting Jupyter Notebook
A gallery of interesting Jupyter Notebooks
CheatSheet jupyter notebook gallery
Jupyter kernels
Jupyter Notebook Server API
Jupyter tutorial at Facultad de Minas (Universidad Nacional de Colombia) in Spanish
JupyterCon 2017 Sprint Day
JupyterCon 2018 Sprint Day
Other Tutorials & Learning Resources
Wiki Cai Dat

<https://github.com/jupyter/jupyter/wiki>

Python and Jupyter notebooks

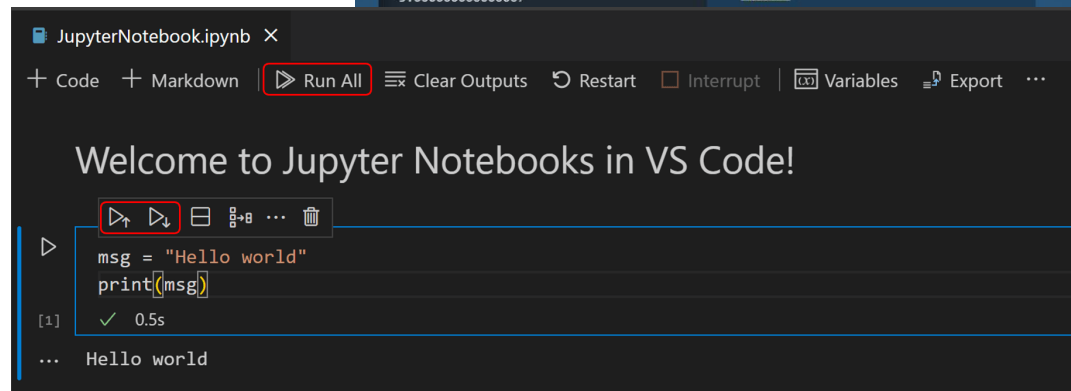
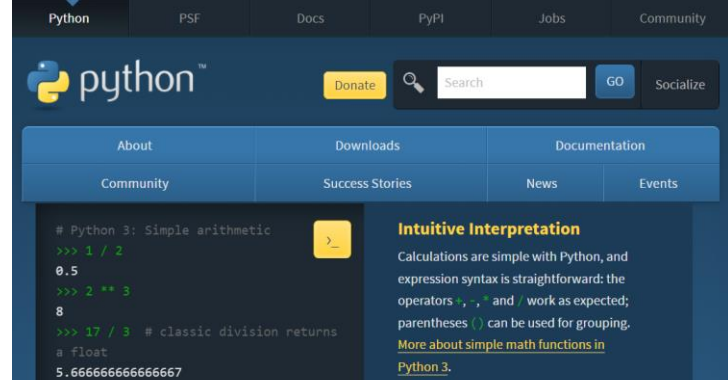


Installation alternatives

1. Google Colab
2. Python and Browser
3. Python and IDE
4. Anaconda Toolkit
5. Miniconda Toolkit

1) Python and IDE

- Installing Python
<https://www.python.org/>
- Choosing an IDE compatible with Jupyter notebooks (e.g., VS Code)
<https://code.visualstudio.com/>
- Local development
- Runs on your computer's resources



2) Python with Anaconda



- A toolkit that includes software to develop Data Science projects
- It includes Python, also the R programming language
- It installs more than 1.500 packages
- Requires around 3GB of space

<https://www.anaconda.com/>



Individual Edition

Your data science toolkit

With over 25 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries.

Anaconda Individual Edition

DataScienceGroup
datascience.dsv.su.se




Stockholm
University

3) Python with Miniconda

- A smaller version of Anaconda (~60MB)
- Install manually the libraries as required.

<https://docs.conda.io/en/latest/miniconda.html>

 Conda

[Docs](#) » [Miniconda](#) [Edit on GitHub](#)

Miniconda

Miniconda is a free minimal installer for conda. It is a small, bootstrap version of Anaconda that includes only conda, Python, the packages they depend on, and a small number of other useful packages, including pip, zlib and a few others. Use the `conda install` command to install 720+ additional conda packages from the Anaconda repository.

[See if Miniconda is right for you.](#)

System requirements

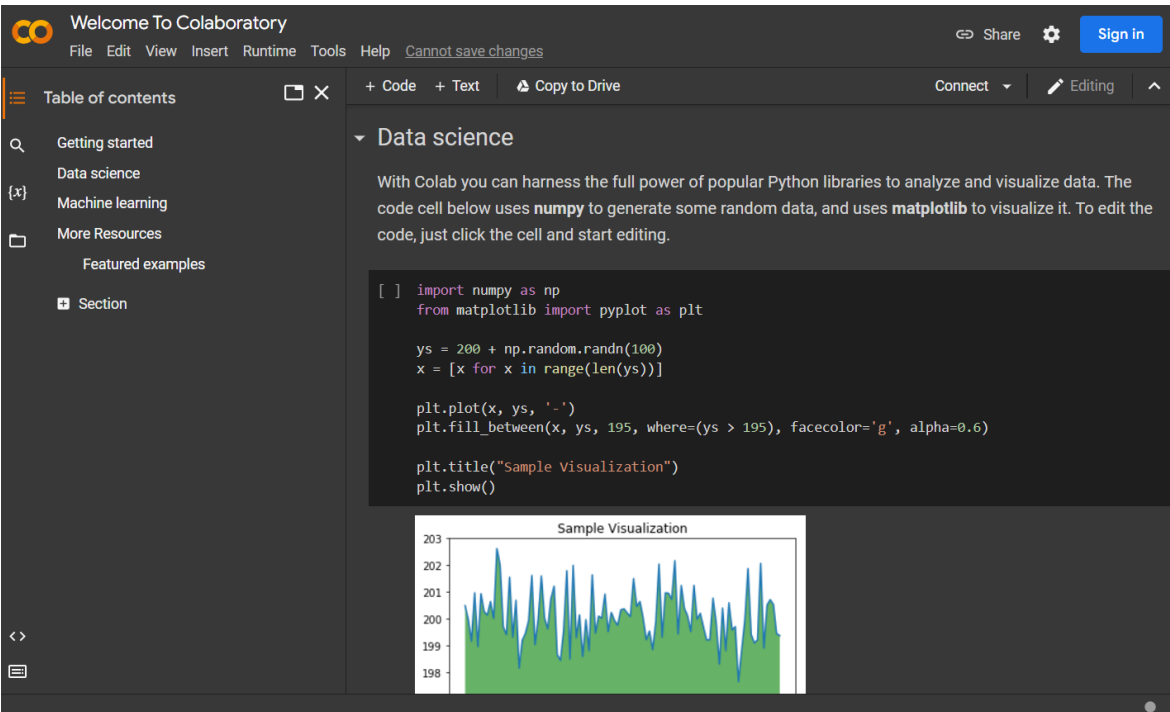
- License: Free use and redistribution under the terms of the [EULA for Miniconda](#).
- Operating system: Windows 8 or newer, 64-bit macOS 10.13+, or Linux, including Ubuntu, RedHat, CentOS 7+, and others.
- If your operating system is older than what is currently supported, you can find older versions of the Miniconda installers in our [archive](#) that might work for you.
- System architecture: Windows- 64-bit x86, 32-bit x86; macOS- 64-bit x86 & Apple M1 (ARM64); Linux- 64-bit x86, 64-bit aarch64 (AWS Graviton2 / ARM64), 64-bit IBM Power8/Power9, s390x (Linux on IBM Z & LinuxONE).
- The `linux-aarch64` Miniconda installer requires `glibc >=2.26` and thus will **not** work with CentOS 7, Ubuntu 16.04, or Debian 9 ("stretch").

DataScienceGroup
datascience.dsv.su.se



Stockholm
University

4) Google Colaboratory



The screenshot displays the Google Colaboratory web interface. At the top, there's a 'Welcome To Colaboratory' header with a 'Sign In' button. Below the header is a menu bar with options like File, Edit, View, Insert, Runtime, Tools, and Help. A sidebar on the left contains a 'Table of contents' and a 'Section' dropdown. The main area is divided into two tabs: '+ Code' (selected) and '+ Text'. The '+ Code' tab shows a code cell with the following Python code:

```
[ ] import numpy as np
    from matplotlib import pyplot as plt

    ys = 200 + np.random.randn(100)
    x = [x for x in range(len(ys))]

    plt.plot(x, ys, '-')
    plt.fill_between(x, ys, 195, where=(ys > 195), facecolor='g', alpha=0.6)

    plt.title("Sample Visualization")
    plt.show()
```

Below the code cell, the output is a line plot titled 'Sample Visualization'. The x-axis represents an index from 0 to 99, and the y-axis represents values ranging from 198 to 203. The plot shows a blue line representing the data points, with a green shaded area underneath it, indicating the region where the values are greater than 195.

- Hosted online
- Runs on Google's servers
- Requires uploading your data to the cloud
- Runs only for a limited time

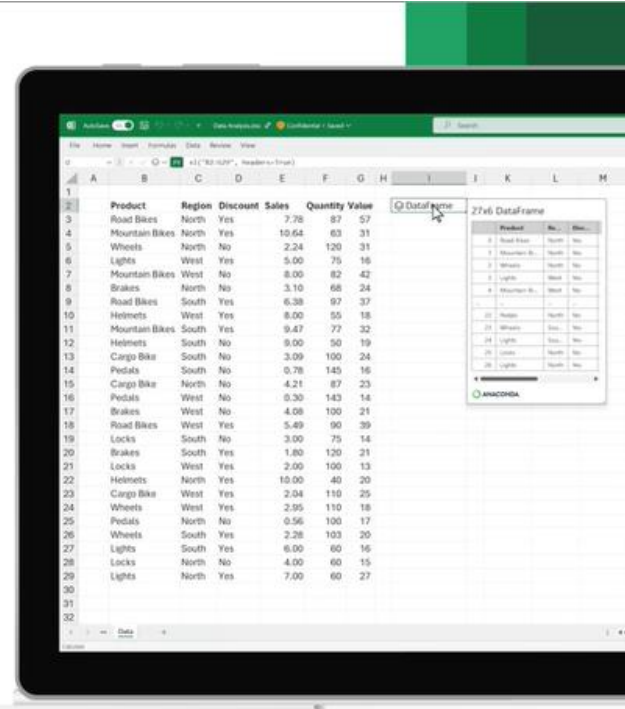
Access from your browser:

<https://colab.research.google.com/>

[Python in Excel \(microsoft.com\)](https://microsoft.com/python/excel)

Microsoft 365

Python in Excel



The image shows a laptop screen displaying Microsoft Excel. The Excel window has a green title bar and a ribbon with tabs for 'File', 'Home', 'Insert', 'Formulas', 'Data', 'Review', and 'View'. The 'Data' tab is active, showing a 'Dataframe' window on the right. The dataframe contains the following data:

Product	Region	Discount	Sales	Quantity	Value
Road Bikes	North	Yes	7.78	87	57
Mountain Bikes	North	Yes	10.64	63	31
Wheels	North	No	2.24	120	31
Lights	West	Yes	5.00	75	16
Mountain Bikes	West	No	8.00	82	42
Brakes	North	No	3.10	68	24
Road Bikes	South	Yes	6.38	97	37
Helmets	West	Yes	8.00	55	18
Mountain Bikes	South	Yes	9.47	77	32
Helmets	South	No	9.00	50	19
Cargo Bikes	South	No	3.09	100	24
Pedals	South	No	0.78	143	16
Cargo Bikes	North	No	4.21	87	23
Pedals	West	No	0.30	143	14
Brakes	West	No	4.08	100	21
Road Bikes	West	Yes	5.49	90	39
Locks	South	No	3.00	75	14
Brakes	South	Yes	1.80	120	21
Locks	West	Yes	2.00	100	13
Helmets	North	Yes	10.00	40	20
Cargo Bikes	West	Yes	2.04	110	25
Wheels	West	Yes	2.95	110	18
Pedals	North	No	0.56	100	17
Wheels	South	Yes	2.28	103	20
Lights	South	Yes	6.00	60	16
Locks	North	No	4.00	60	15
Lights	North	Yes	7.00	60	27

DataScienceGroup
datascience.dsv.su.se

Checklist for upcoming Lab 1

- ☐ Install Python >3.9 in your personal computers
- ☐ Install these packages with **pip**
(or **conda** if using Anaconda)
 - ☐ jupyter
 - ☐ numpy
 - ☐ pandas
 - ☐ matplotlib
- ☐ Open and run the Jupyter notebook Lab 0 available from the iLearn website
- ☐ **Optional:** Practice the basic syntax of Python

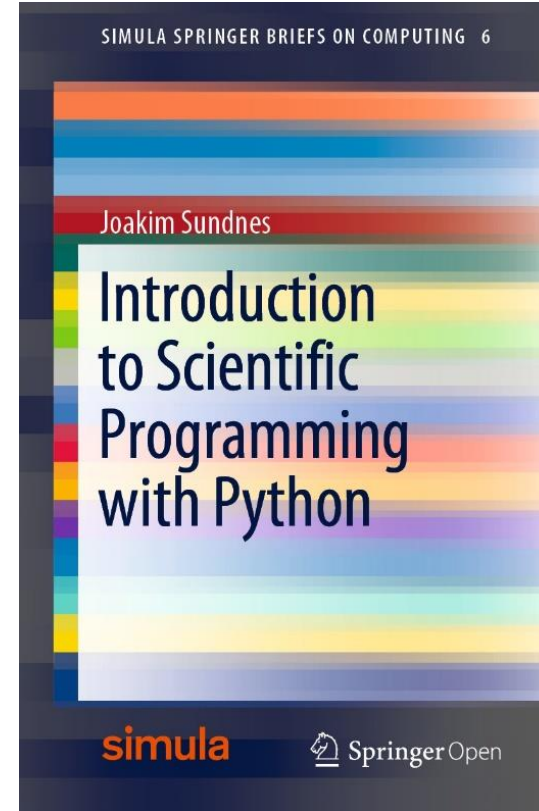
Python prerequisites for the labs

(videos will be posted on iLearn explaining these concepts)

1. Pythonic syntax
 - Interpreter, Math Operations, Variables, Function Call
2. Data Types and Data Structures
 - Variables, List, Tuples, String, **Dictionary!**
3. Package Management (PIP)
 - numpy, pandas, jupyter, scikit-learn, scipy
 - Working with existing open-source projects
4. Conditionals and Loops with Jupyter
5. Definition of Functions

Resources

- [Introduction to Scientific Programming with Python | SpringerLink](#)
- [Beginners Guide / Programmers - Python Wiki](#)



ILOs Recap

- I. Assess the **quality of a structured dataset**
- II. Formulate **analytical questions** that can be solved with descriptive and exploratory methods
- III. Recognize the types of questions that can be answered with **predictive methods**
- IV. Configure **Python** to design and implement a data science project



Luis Quintero



luis-eduardo@dsv.su.se



<https://luisqtr.com>



DataScienceGroup
datascience.dsv.su.se



Stockholm
University