Preparation

- Install „Virtual Box“
- Go to http://hortonworks.com/downloads/#sandbox
  - Search for „Archive“ and download 2.4 version - if you have less than 10GB memory
- OR: search for „Tutorial: Sandbox on Azure“ and follow instructions
- Start virtual machine and open
  - http://127.0.0.1:8080
  - http://127.0.0.1:9995

FORSCHUNGS CAMPUS

öffentlich-private Partnerschaft für Innovationen

SPONSORED BY THE

Federal Ministry of Education and Research

Omics-based Cancer Diagnostics

Tim Conrad, Freie Universität Berlin

Forschungs Campus MODAL, **MedLAB**

Introduction

SPARK & HDFS

Getting Started

Simple Example

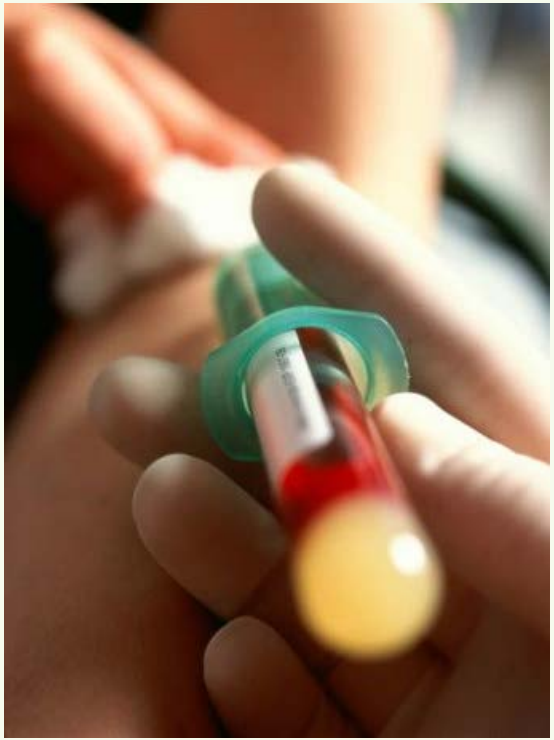
Application

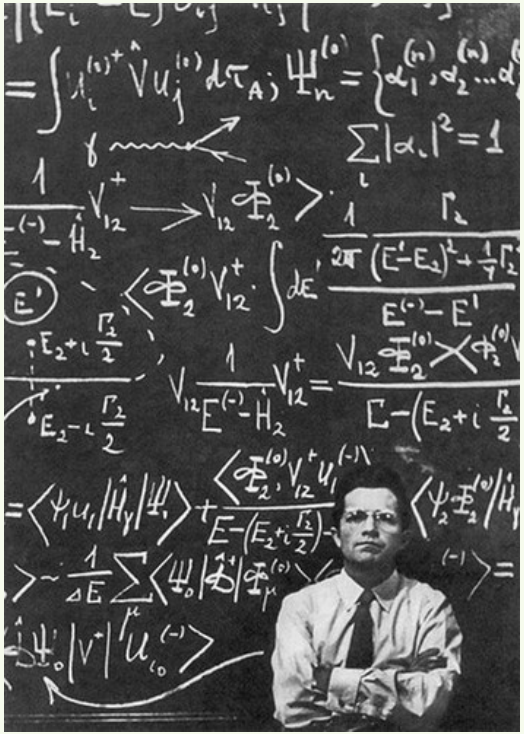
Classification of biomedical data

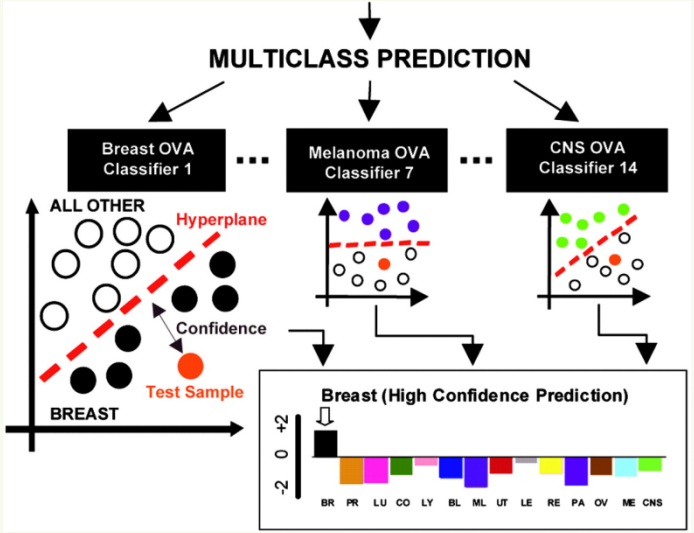
ACQUIRE DATA

ANALYSE DATA

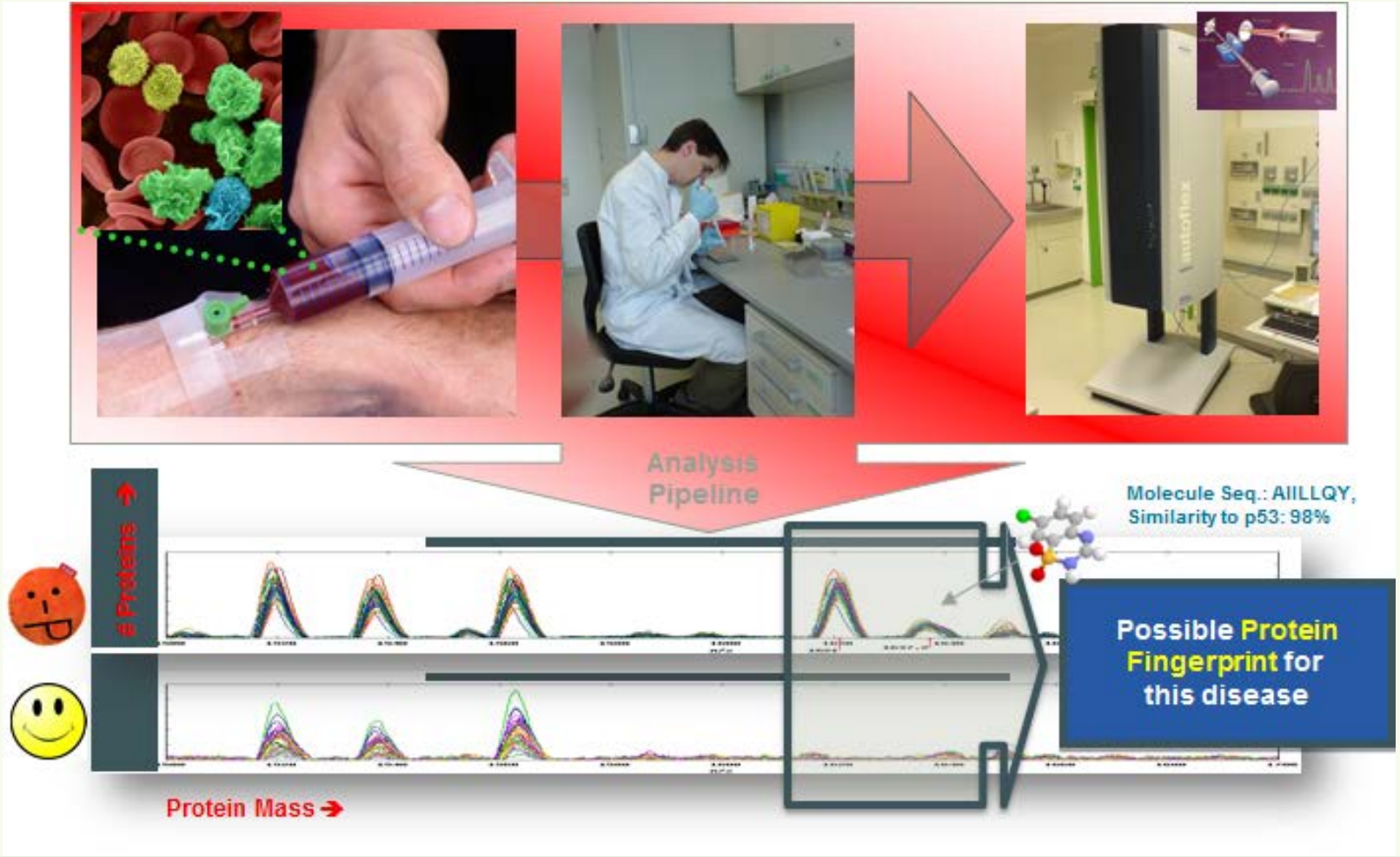
CLASSIFY DATA





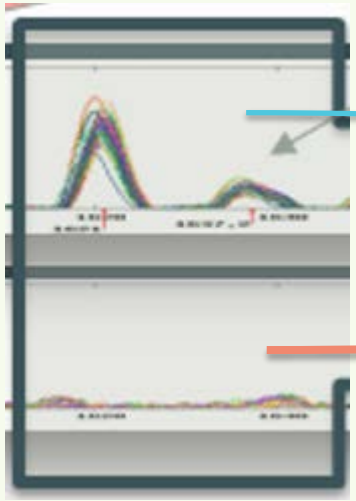


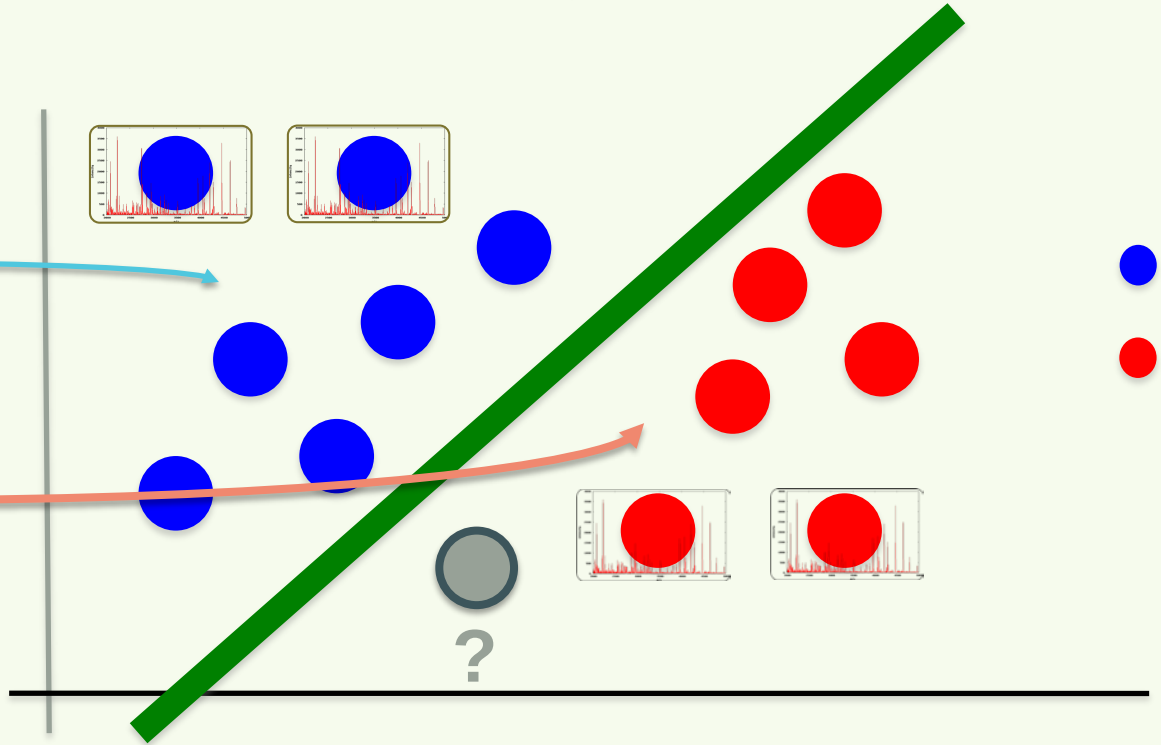

GOAL


Analysis Pipeline
Proteins
Protein Mass
Molecule Seq.: AIILLQY, Similarity to p53: 98%
Possible Protein Fingerprint for this disease

Classification (e.g. SVM)

Determine separating line

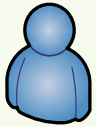


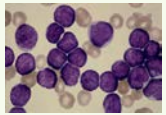


Healthy patients

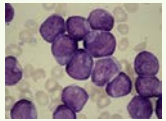
Diseased Patients

Biomarkers Discovery Workflow

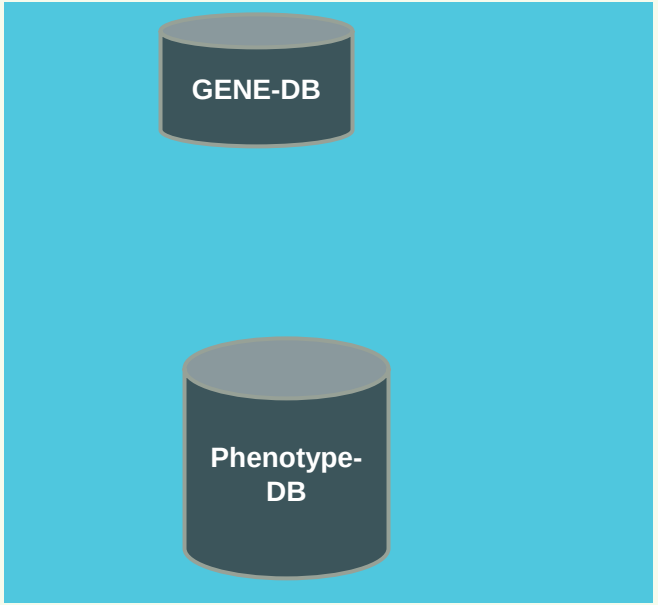


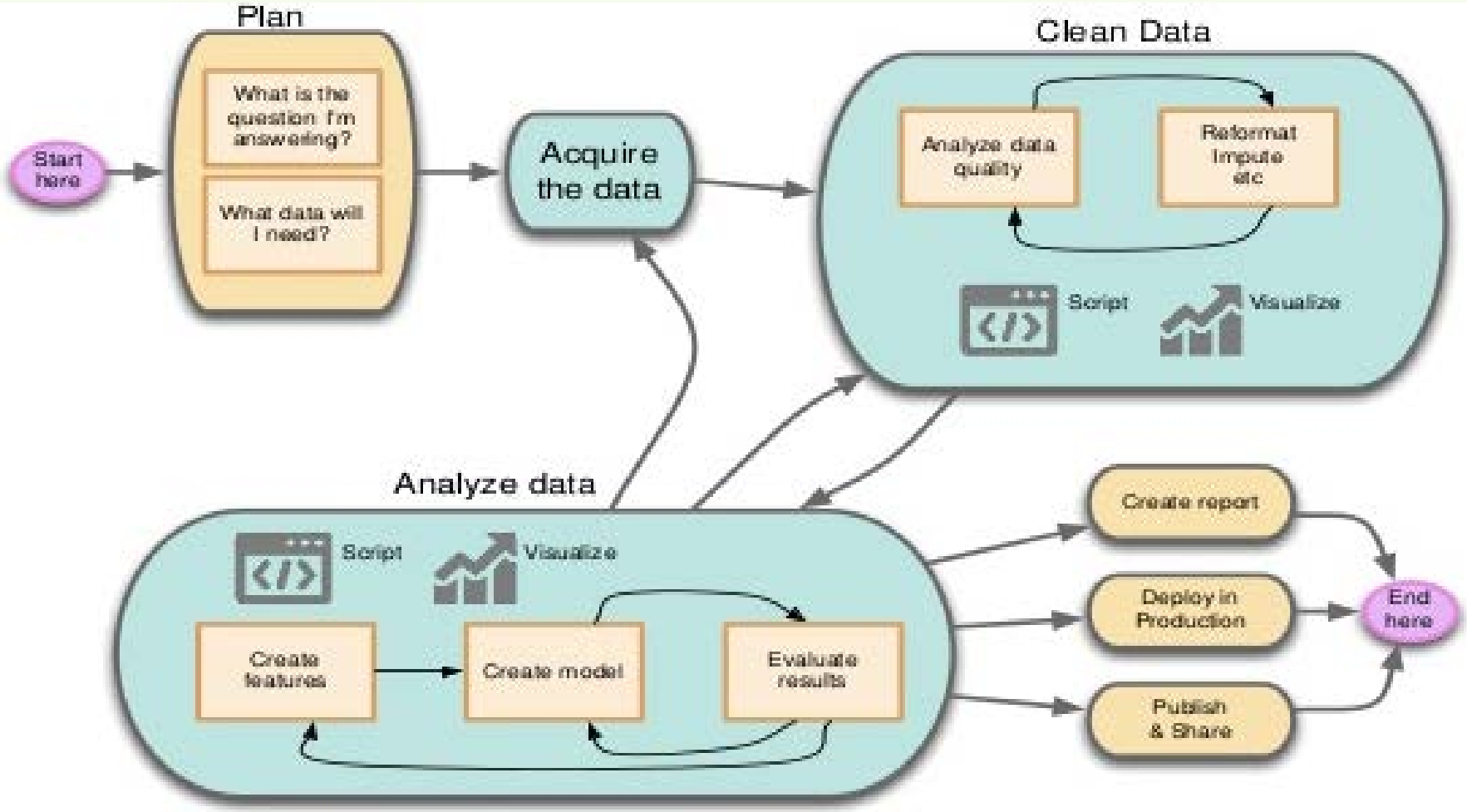


Disease



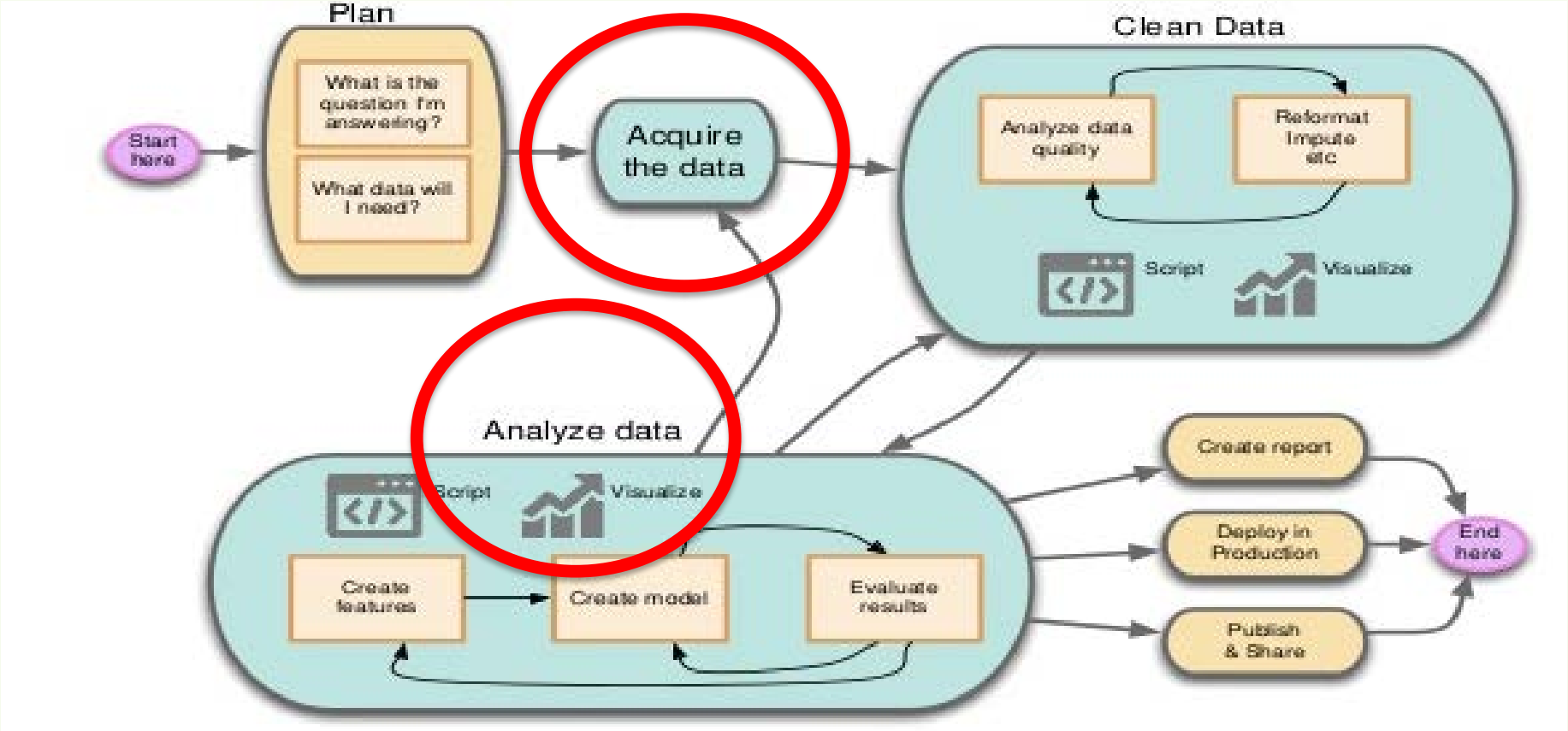
Normal



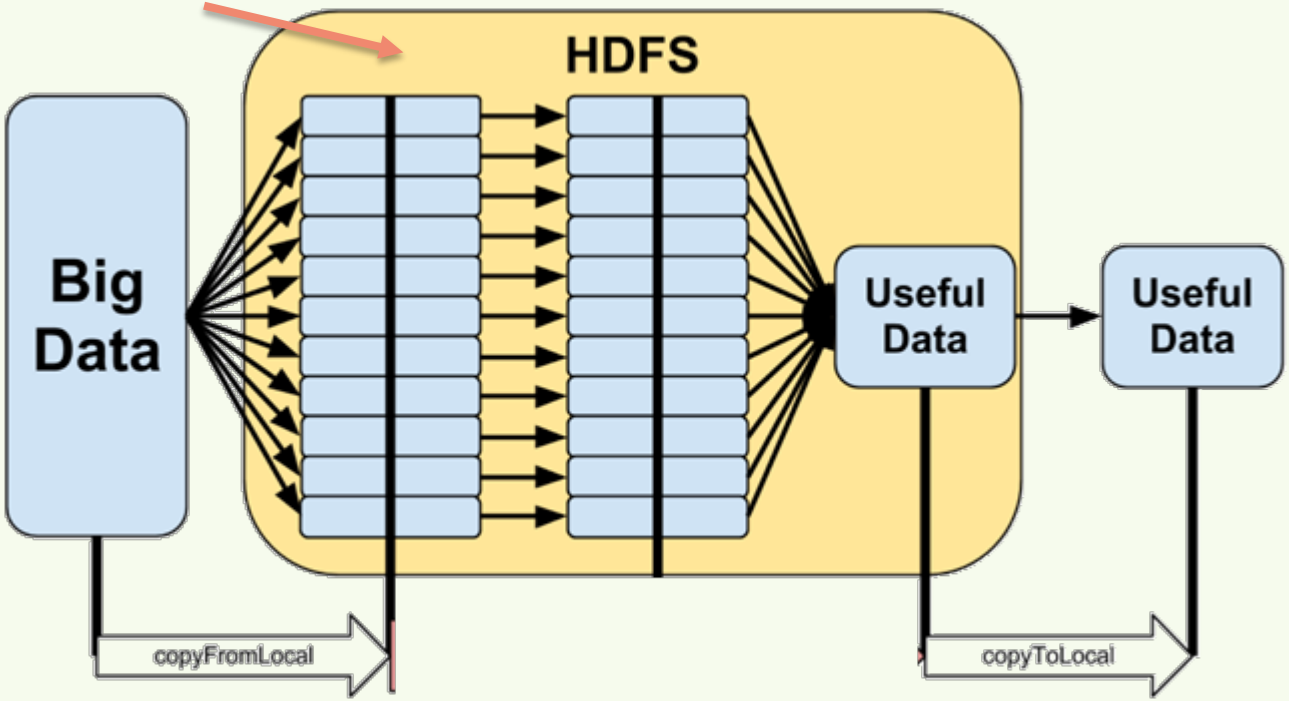

Plan
What is the question I'm answering?
What data will I need?
Start here
Acquire the data
Clean Data
Analyze data quality
Reformat Impute etc
Script
Visualize
Analyze data
Script
Visualize
Create features
Create model
Evaluate results
Create report
Deploy in Production
Publish & Share
End here


PROBLEMS
AHEAD
CAUTION

What if this is too big for a single machine?



Split data AND analysis to many machines!



Machine learning with Spark

What do we learn?

- How to use a LOCAL SPARK system with HDFS
- How to change to a DISTRIBUTED SPARK system
- Examples:
  - k-Means
  - Predicting breast cancer from proteomics data

jupyter notebook_test Last Checkpoint: Last Wednesday at 11:56 AM (autosaved)

File Edit View Insert Cell Kernel Help Python 3

Markdown Cell Toolbar: None

Testing IPython Notebooks from MarkDown files

In [6]:
```
>>> print("foo")
>>> l = [i*i for i in range(100)]
```

```
foo
```

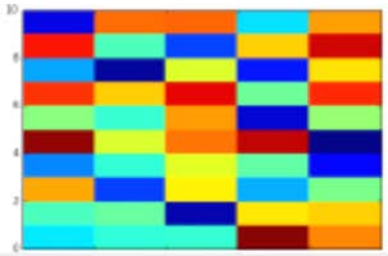
In [7]:
```
>>> %matplotlib inline
```

In [8]:
```
>>> import matplotlib
>>> import numpy as np
>>> import matplotlib.pyplot as pl
```

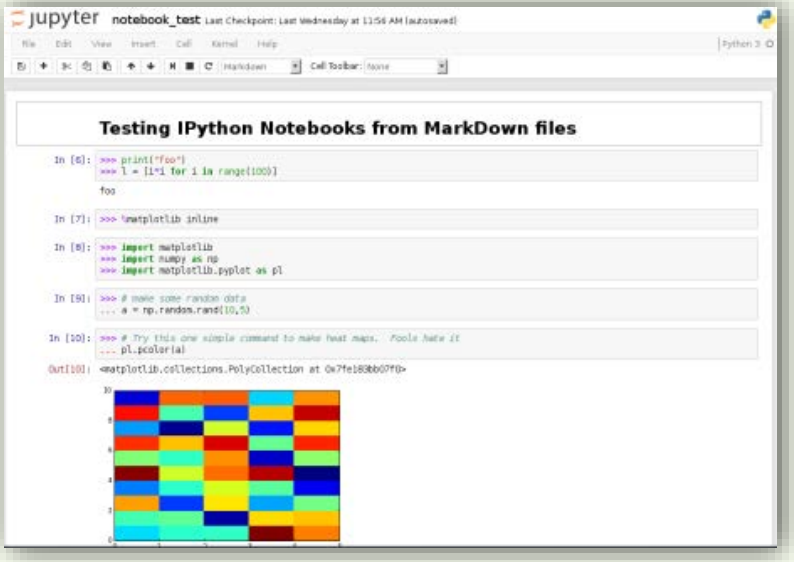
In [9]:
```
>>> # make some random data
... a = np.random.rand(10,5)
```

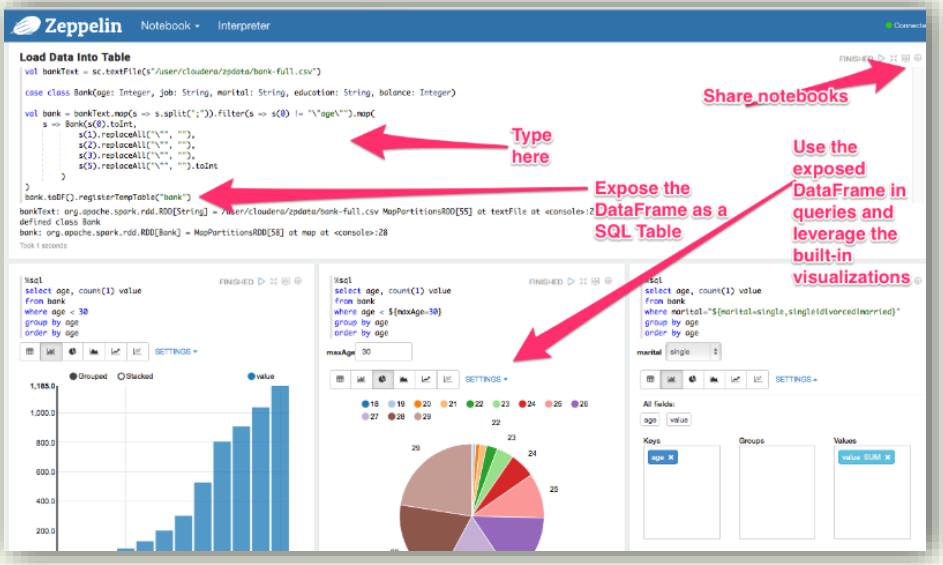
In [10]:
```
>>> # Try this one simple command to make heat maps.  Fools hate it
... pl.pcolor(a)
```

Out[10]:
```
<matplotlib.collections.PolyCollection at 0x7fe1893bb07f0>
```



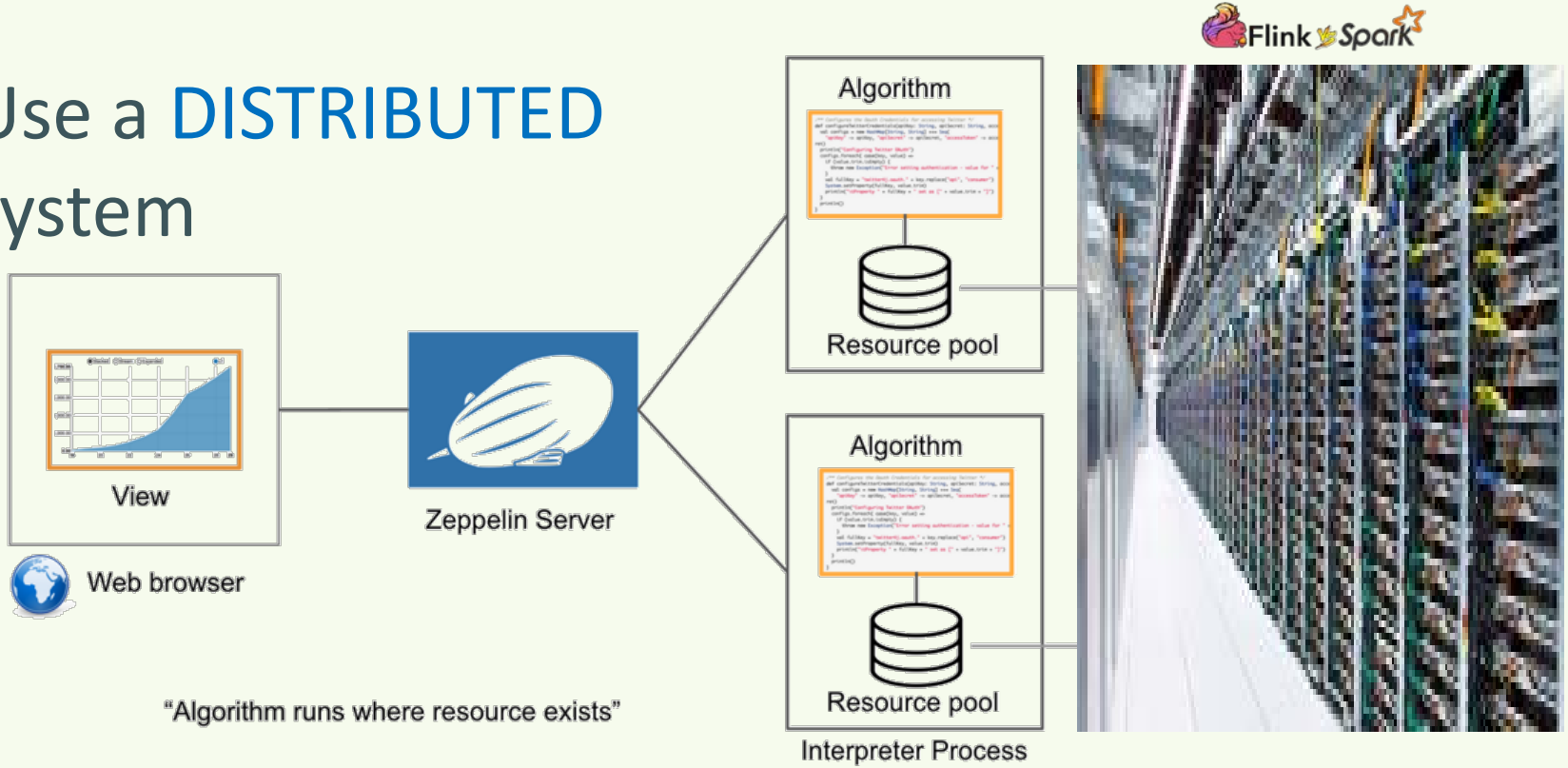
- Cassandra
- Elasticsearch
- Flink
- Geode
- Hive
- Ignite
- Lens
- Markdown
- Postgresql, Hawq
- Scalding
- Shell
- Spark
- Tajo
- Build your own !


jupyter
notebook_test
Testing IPython Notebooks from MarkDown files


Zeppelin
Notebook
Interpreter
Load Data Into Table
Share notebooks
Type here
Expose the DataFrame as a SQL Table
Use the exposed DataFrame in queries and leverage the built-in visualizations

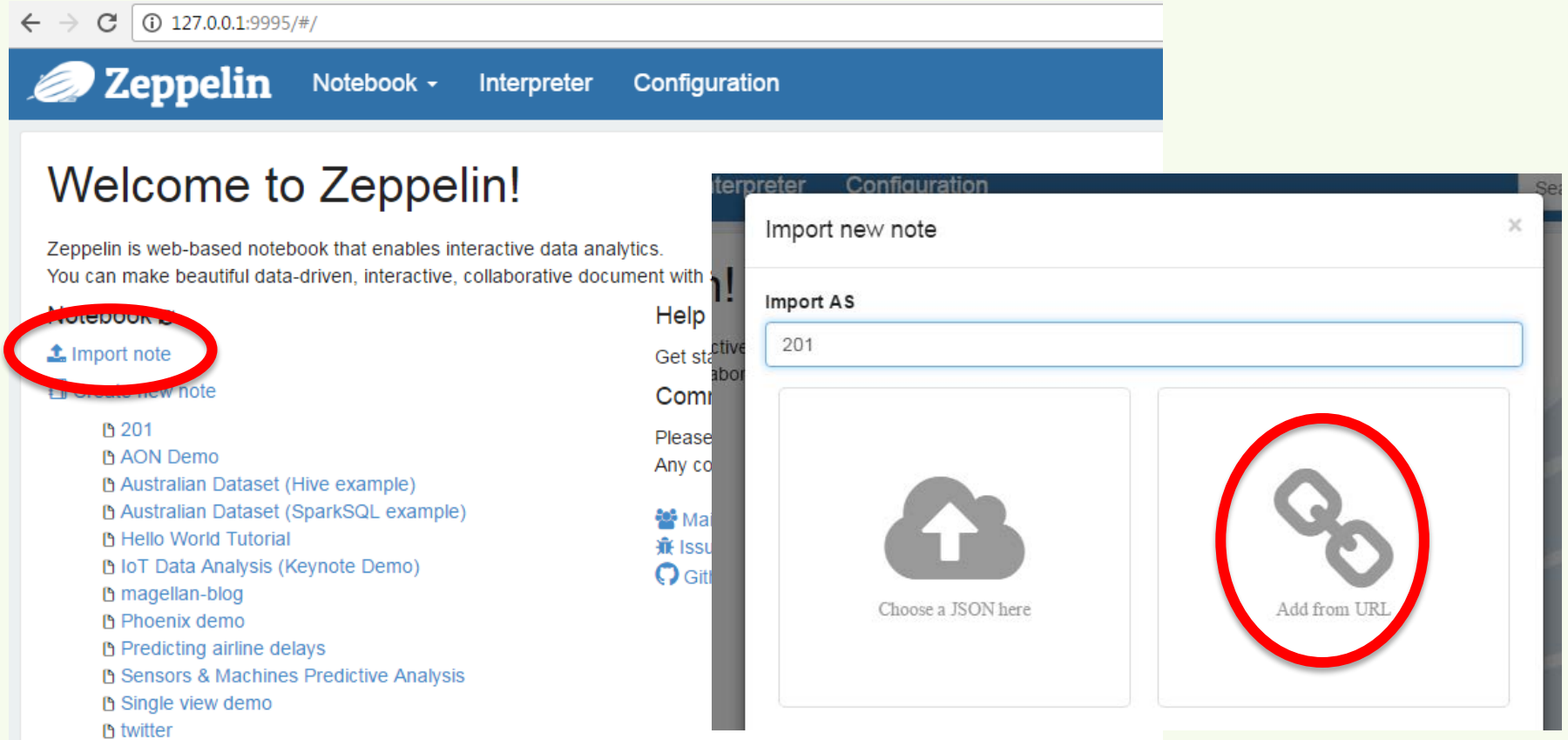
What can be done next?

- Use a DISTRIBUTED system



Hands-on

- Task 1 (201): bit.ly/dsss201
  - Diagnose breast cancer patients
- Task 2 (101): bit.ly/dsss101
  - Get more familiar with Spark & Zeppelin



1. Open IN YOUR BROWSER: bit.ly/dsss201
2. Copy new link
3. Use new link for importing