# Data Analytics 2

Summer Term 2019

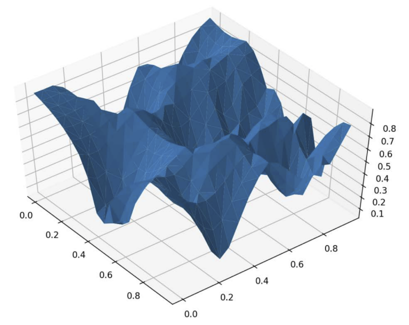
Dr. Pascal Kerschke, Dennis Assenmacher,
Moritz Seiler, Raphael Prager, Matthias Carnein

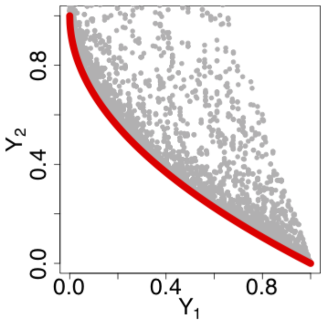
Information Systems and Statistics,
University of Münster, Germany

# Case Study Introduction

# Outline

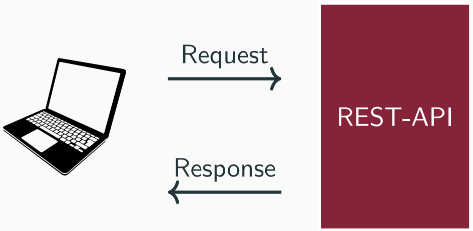
- The task: **function approximation (regression) + multi-objective optimization**
- Deliverables and assessment criteria
- Group building





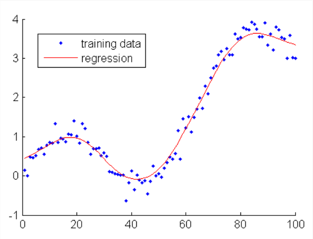
# Step 1: Function Approximation

- We put regression *(function approximation via supervised learning)* and multi-objective optimization together
- Every group gets a **specific, unknown multi-objective problem**
  - The problem consist of **two separate functions** *(minimization!)*
- Via a web service, you can request a limited *(2000)* number of points
- Run a **machine learning algorithm** on observations from both functions to approximate them *(surrogate model)*


Request

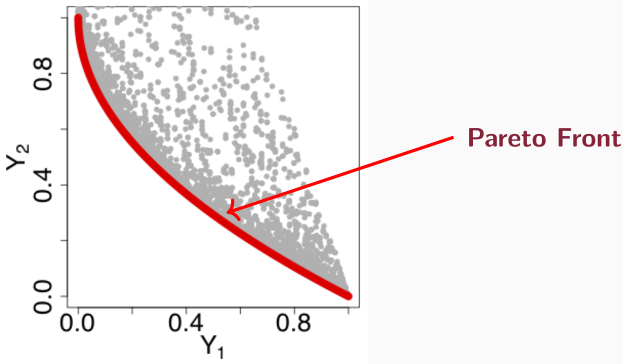
Response

REST-API



# Step 2: Optimization

- Find the approximated **pareto front for both surrogate models** *($Y_1, Y_2$)*
- The functions are in 3D, in the unit hypercube $[-5, 5]^3$
- Evolutionary algorithms are allowed, but not enforced
- **Use your budget wisely**



**Pareto Front**

# Task Summary

Your tasks:

1. Generate good surrogate models for **both functions**. *(Online)* *(any supervised learning technique can be used)*
2. Use these models to **approximate the pareto front** *(Offline)*
3. Learn something about the modeled problem *(try visualization!)*

You may:

- **First try out your methods in test mode** *(unlimited sampling)*
- Try out and assess different approaches/algorithms
- Use several models

You shall:

- Attempt rapid prototyping, take care of parameters
- **Discuss within the group and also between groups**

# Deliverables

We expect you to deliver a short written report *(***not more** *than 10 pages)*, containing:

- Which **techniques** have you considered for the modeling
- Why did you finally choose one, and how did you set **parameters**?
- What kind of **optimization algorithm** did you use and why?
- How did you integrate optimization and modeling, how did you **spend your budget** and why?
- What have you learnt **about your problem**? How did you draw these conclusions *(reasoning)*?
- Please provide at least **20 of your optimal points** that you have found *(reasoning)*

# Important Dates

June 5 Begin of the case study and distribution of access tokens.

July 9 Final submission of the **written report**, the **presentation slides** and your **(commented) code** through the learnweb. (July/09/2019 11:59pm)

July 10-11 Final presentations (3 each day; 20 minutes). The presenting group will be randomly drawn without replacement.

Within the presenting group the **presenter** will also be chosen **randomly** (one group = one presenter). All group members have to be present at **both** presentation days.

# Client

- You can **access** your data via a provided **client** (learnweb)
- Each group gets a **unique token**
- We offer three different endpoints:
  1. `api-test2D`
  2. `api-test3D`
  3. `api`
- The first two endpoints are for **testing purpose**. They can be accessed without restrictions!
- Each endpoint has **two functions**: `1,2`
- The **third endpoint** are your groups **individual functions** that needs to be analyzed. **Access is restricted** to 2000 observations **for both functions** combined!

!! Please do not fetch more than **50 observations per request** !!

# Client Example

- `apirequest(input, func, endpoint)`

  `input`: data frame with two or three features and up to 50 observations

  `func`: either 1 or 2 *(defines which function shall be accessed)*

  `endpoint`: "api-test2D", "api-test3D" or "api"

- We want to get the **second 3D test function evaluation** for two points:
  1. The vectors are defined as: $(2, 3.4, 2)^T, (0.4, -0.5, 0.4)^T$
  2. Next, define a data frame: `input = data.frame(x=c(2,0.4), y=c(3.4, -0.5), z=c(2,0.4))`
  3. Observe the result: `response = apirequest(input, 2, "api-test3D")`
  4. `response` is a data frame containing the **target values of function** 2

# A few hints to get a good grade

Do NOT **waste your budget**!

- Test your approaches by **utilizing** the test functions
- Do NOT build your models on 1000 random data points each *(be smarter)*

Try **different approaches**:

- Supervised Learning methods *(regression tree, ANN's, etc.)*
- Optimization algorithms *(EA's)*
- There are **a lot of algorithms** out there that we did not cover...

# A few hints to get a good grade

Work together:

- **Meet** with your group and **discuss** different approaches
- Make **plans and timetables**, distribute work
- Peer programming...

Provide a **good documentation**

- Comment your code!!!
- **Make notes** about test runs *(experimental journal)*
- Use LaTeX *(Overleaf)*

# If you have problems...

Ask the experts:

Moritz Seiler

Raphael Prager

- Each Wednesday
- Between 10.15 and 10.45 AM (and longer if necessary)
- Leonardo Campus 18