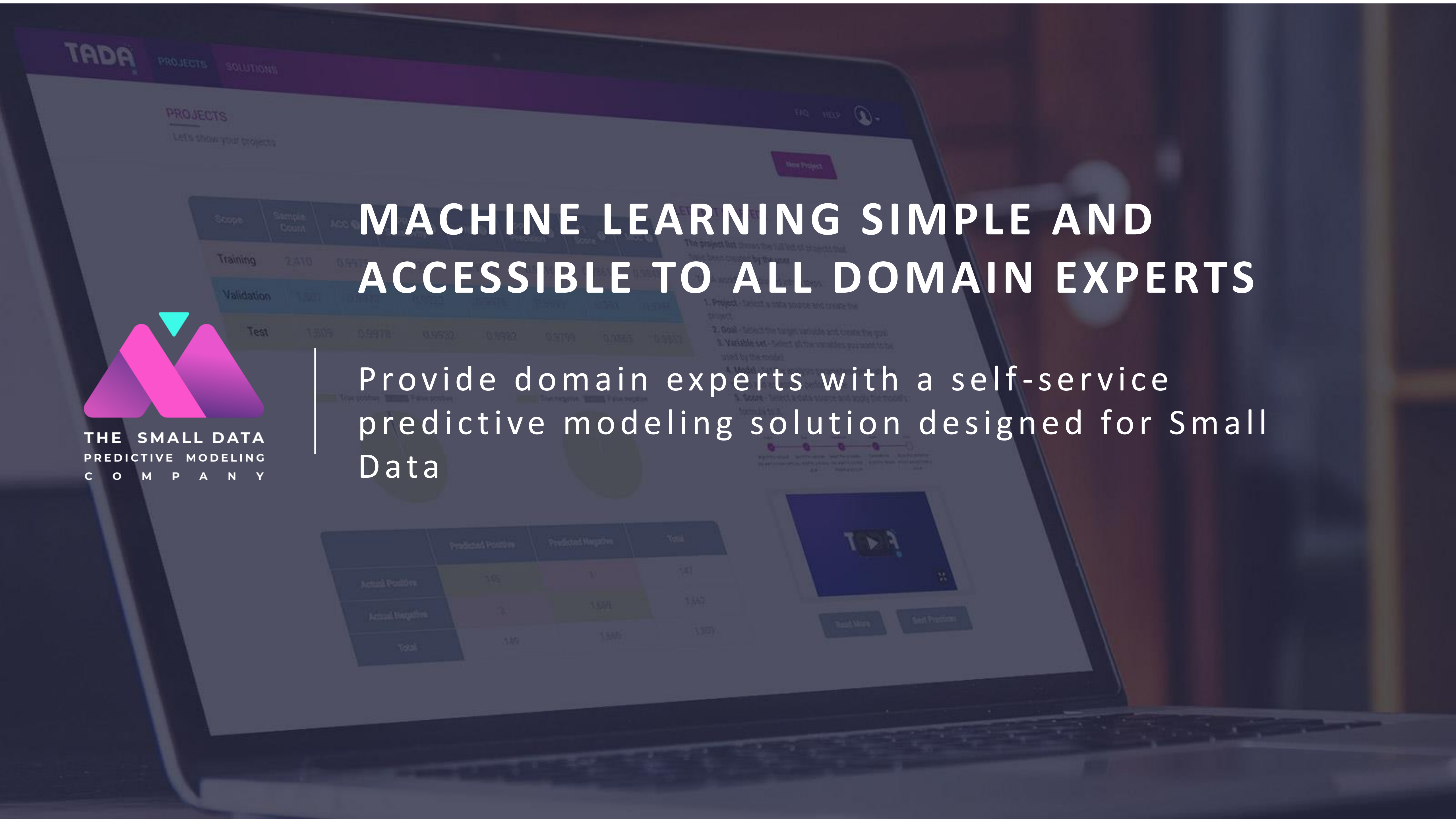


MACHINE LEARNING SIMPLE AND ACCESSIBLE TO ALL DOMAIN EXPERTS

THE SMALL DATA
PREDICTIVE MODELING
COMPANY

Provide domain experts with a self-service predictive modeling solution designed for Small Data www.mydatamodels.com

635 Route des Lucioles
06560 Valbonne, Sophia Antipolis
France

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

Inside TADA: a peek inside MyDataModels Platform for Small Data analysis and prediction

Data Science Meetup - Nice - Sophia-Antipolis

Nice, France
704 membres · Groupe public
Organisé par **Data Science meetup N.**

Partager :

Dr Carlo Fanara

Head of Research
T. +33 650178 054

MYDATAMODELS
The Automated Machine Learning Company
www.mydatamodels.com
635 Route des Lucioles
06560 Valbonne, Sophia Antipolis
France

Inside TADA: a peek inside MyDataModels Platform for Small Data analysis and prediction

Summary

- Introduction on MyDataModels and TADA
- Notes on Evolutionary Algorithms
  - Foundations and Nomenclature
  - Genetic Operators
  - Applications (when and where)?
  - Open source frameworks and tools: Examples
  - Bibliography

Data Science Meetup - Nice - Sophia-Antipolis

Nice, France

704 membres · Groupe public

Organisé par Data Science meetup N.

Partager :

Inside TADA: a peek inside MyDataModels Platform for Small Data analysis and prediction

Motivation and Outlook

- Since its foundation, MyDataModels (MDM), has been using **small data**, showing how these ‘democratize’ the field, allowing Domain Experts and professionals to access machine learning results in an unprecedented way.
- Machine Learning (ML) models can be generated with artificial neural network (ANN), deep learning (DL), but also – like in the case of MDM – using **Evolutionary Programming** (EP) and **Genetic Algorithms** (GA).
- Whilst the latter may occasionally cost in terms of execution time, ways to define early convergence can be found. This is one of the efforts currently undertaken at MDM: and it is worth, as the outcome is given with mathematical formulae which include the variables from the original dataset. Thus, **models become *explainable***. And, since a model trained on a machine can be loaded on an ***edge* device**, where it is used to *infer* novel results, **models become *exploitable.***
- After a brief introduction about MyDataModels (MDM), I will cover some essentials about ‘Genetic Algorithms’, illustrating some of the issues and amelioration on which we are working.

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Introducing MyDataModels & TADA

Who are we?

MYDATAMODELS
THE SMALL DATA PREDICTIVE MODELING COMPANY

- Resulting from 10 + years of research in evolutionary algorithms
- Founded on March 2018 in Sophia Antipolis
- Team of 30 peoples with various profiles (Data Science PhDs, Software Engineers, Architects, UX-UI Design, Marketing, Sales, Communication, Management)

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

A strong scientific base: 10+ years of research in evolutionary algorithms

DS TIMELINE

Context: MDM launches TADA Product

02/19

Carlo joins MDM

Diving into the DS topics (docs, benchmarks...)

03/19

Norbert joins the unit!

The dev capabilities to set up the LAB and contribute to the Data Science agenda

04/19

Benches

First tests on benches
Analysis of ZGP docs (real and synth datasets, the ZGP engine...

05/19

Jonathan joins in!

...to accompany the potential customers in their journey towards TADA, but on R&D as well

Release of first round of benches (real datasets)

Jupyter on Z8

06/19

Grip on ZGP ("Z_FORGE")

Multi-session effort to get the grip on the ZGP engine code and prepare a reference document – on the python version

Board for the tasks management is set up (Gitlab)

Gitlab up for code updates

Use cases:
data analysis,
model creation,
prep-processing

07/19

Aurélie joins MDM!

Gear up the research capabilities;
Take care of the general research tasks on ZGP

Revision of synth data docs

Chat with Pat O' Neall

Release of 2nd round of benches (Real Datasets)

08/19

Launch the ZGP tasks

From the ZGP doc get the main tasks:
1. hyperparameters,
2. Operator selection,
3. Variable selection

09/19

Novel activities

State of the art on novel EU challenges (proposals)

The tasks:
- Hyperparameter becomes 'early stop'
- Variable selection becomes validation of existing (wrt state of the art)

10/19

Alexis is back!

Novel activities

Data Analysis
And Use cases

ZGP monitoring (the lab) is functional and used to guide R&D

Context: INRIA-MDM Workshop

t

FROM START, TO DATE

DS unit: Aurelie Boisbunon, Jonathan Daeden, Carlo Fanara, Norbert Leon, Alexis Vighi … and growing!

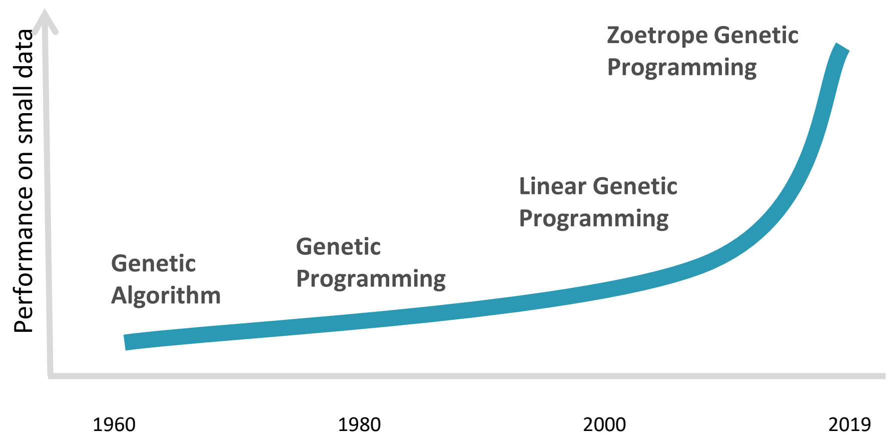
MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

A strong scientific base: 10+ years of research in evolutionary algorithms

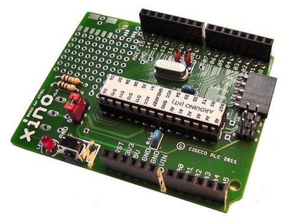
Unique mathematical engine for SMALL DATA

- Based on Evolutionary algorithms, Stochastic & mathematical approach
- Developed for **Small Data** vs statistical & traditional Machine Learning approach
- Easy to **interpret** and **operate**: Models are mathematical expressions



Automated ML platform & EMBEDDED technology

- No specific IT equipment needed: few computing, storage and energy resources
- Runs on **any device**: cloud, private cloud, desktop, laptop, mobile phone and tablets, Edge server and Microcontroller
- Thanks to their **light weight** our model can be **embedded** into onboarded devices

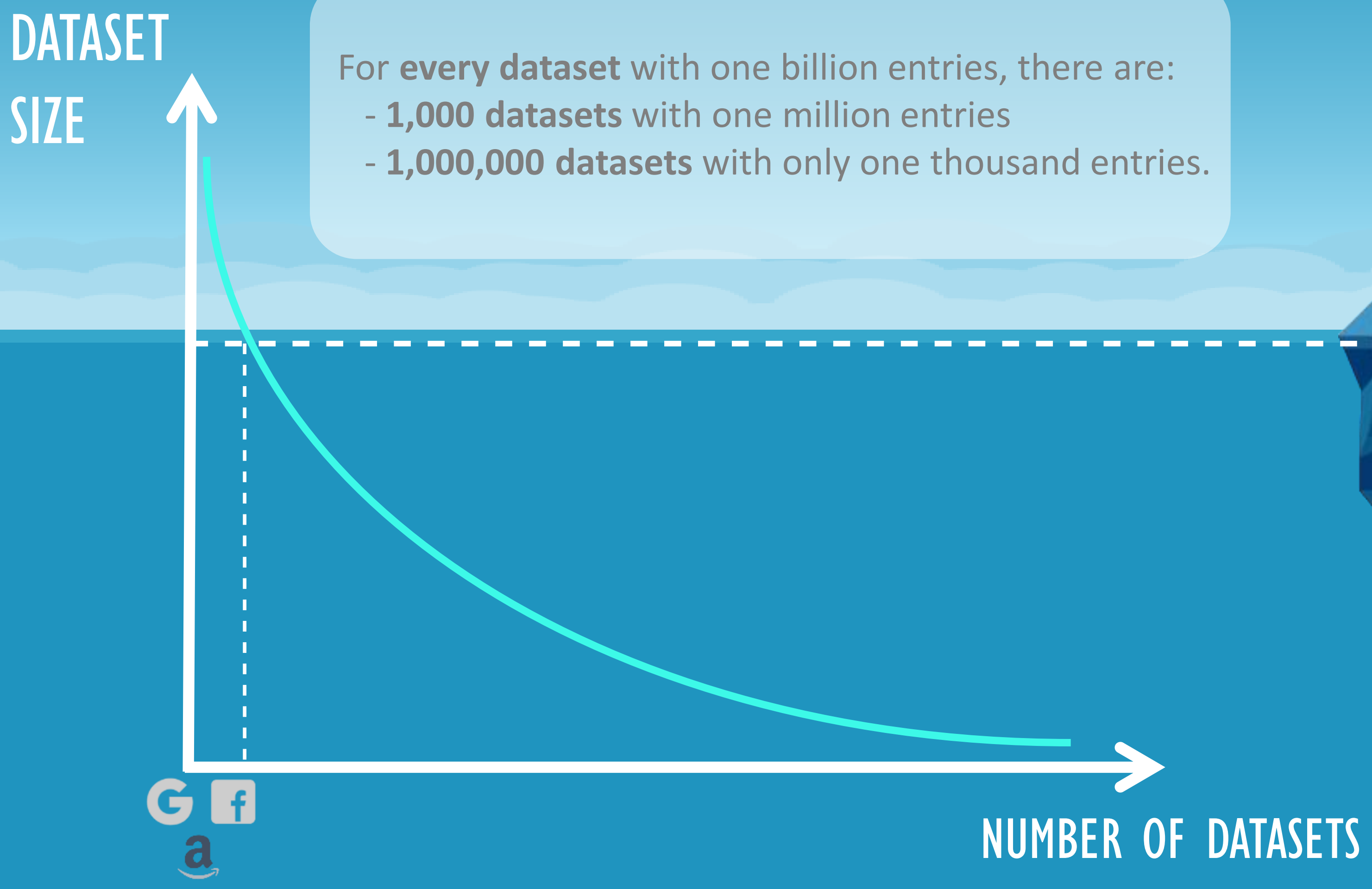


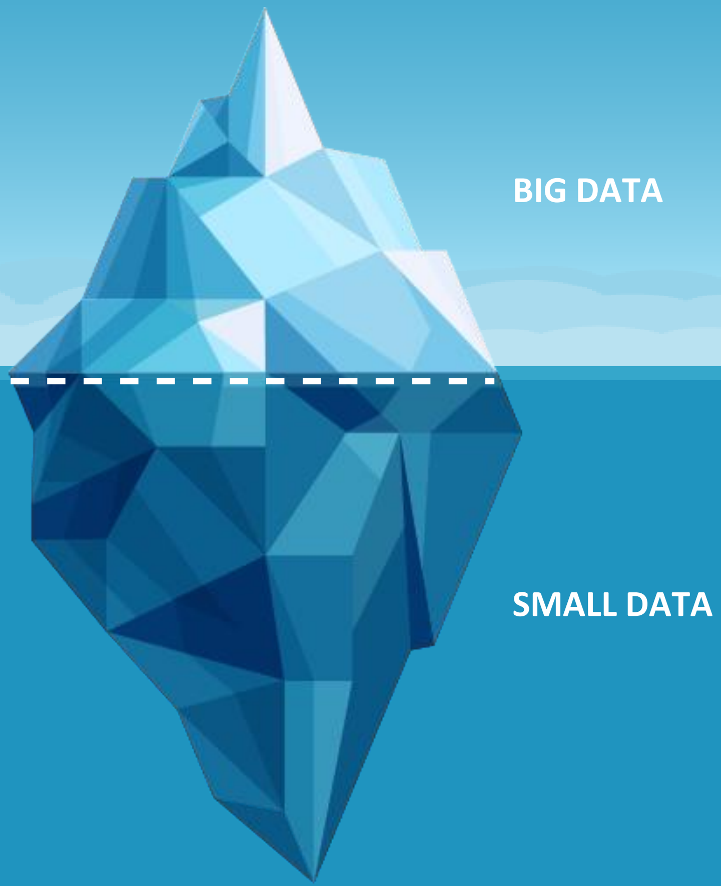
MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Few players with BIG DATAsets & many with SMALL DATAsets

For **every dataset** with one billion entries, there are:
- **1,000 datasets** with one million entries
- **1,000,000 datasets** with only one thousand entries.

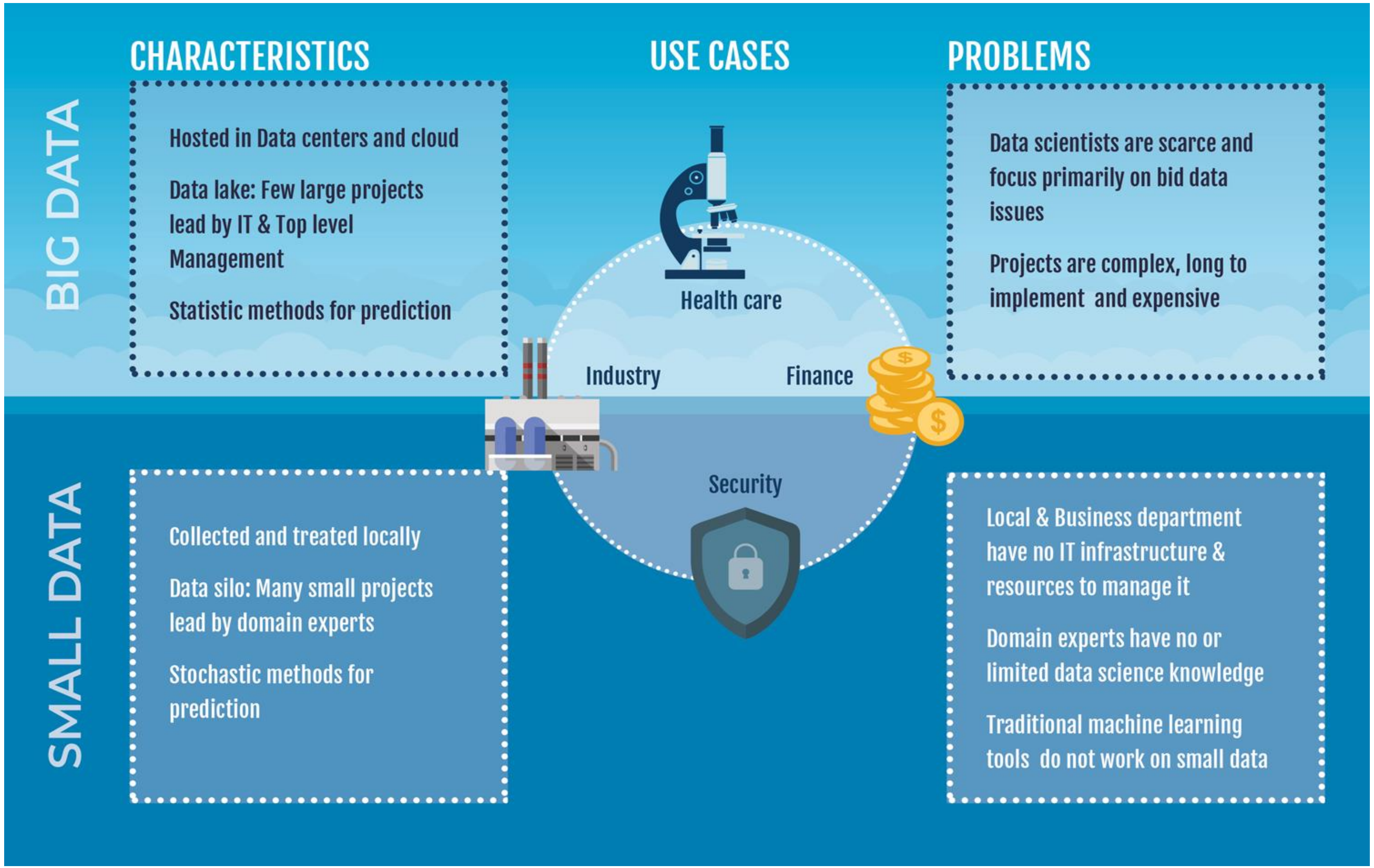




MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Shortage of data scientists & isolated small datasets prevent large scale adoption of ML


CHARACTERISTICS

USE CASES

PROBLEMS

BIG DATA

Hosted in Data centers and cloud

Data lake: Few large projects lead by IT & Top level Management

Statistic methods for prediction

Data scientists are scarce and focus primarily on bid data issues

Projects are complex, long to implement and expensive

SMALL DATA

Collected and treated locally

Data silo: Many small projects lead by domain experts

Stochastic methods for prediction

Local & Business department have no IT infrastructure & resources to manage it

Domain experts have no or limited data science knowledge

Traditional machine learning tools do not work on small data

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

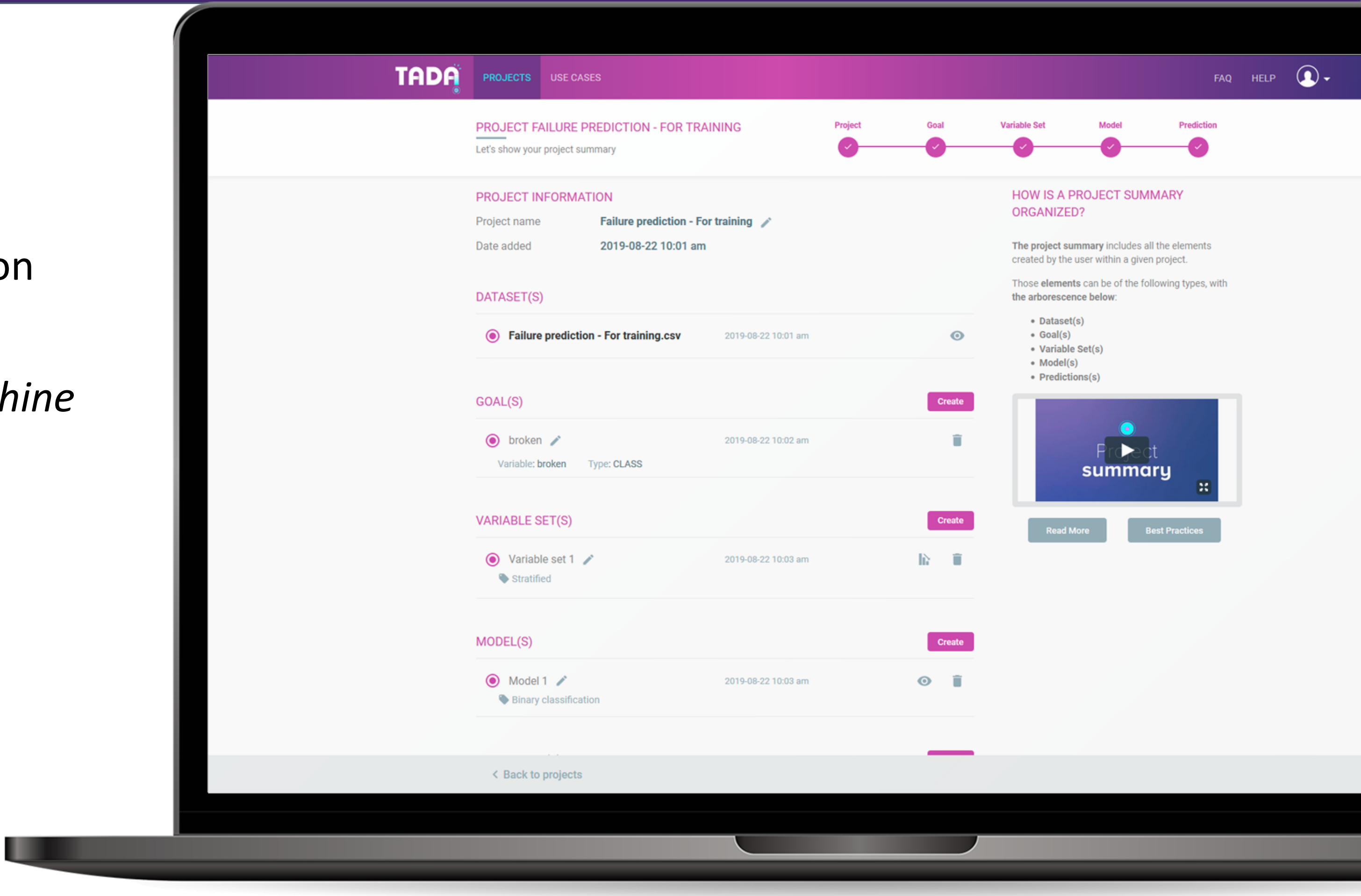
MDM: what we do

TADA: our product

ZGP Engine : ‘Zoetrope’ Genetic Program, based on *Evolutionary programming*

- generate and run *automated* predictive models on client's *own data*
- Easy to use without coding or *background in machine learning*.
- performs well on *Small Data*.

As opposed to Big Data, ‘**Small’ Data** involve ‘small’ samples , ~ hundreds or fewer’ observations…
Ok, ‘small’ fits on a laptop, but ‘ill defined’



PROJECT FAILURE PREDICTION - FOR TRAINING
Let's show your project summary

PROJECT INFORMATION
Project name Failure prediction - For training
Date added 2019-08-22 10:01 am

DATASET(S)
Failure prediction - For training.csv 2019-08-22 10:01 am

GOAL(S)
broken 2019-08-22 10:02 am
Variable: broken Type: CLASS

VARIABLE SET(S)
Variable set 1 2019-08-22 10:03 am
Stratified

MODEL(S)
Model 1 2019-08-22 10:03 am
Binary classification

HOW IS A PROJECT SUMMARY ORGANIZED?

The project summary includes all the elements created by the user within a given project.

Those **elements** can be of the following types, with **the arborescence below**:

- Dataset(s)
- Goal(s)
- Variable Set(s)
- Model(s)
- Predictions(s)

(*) ‘Zoe…’ what? One of coming slides

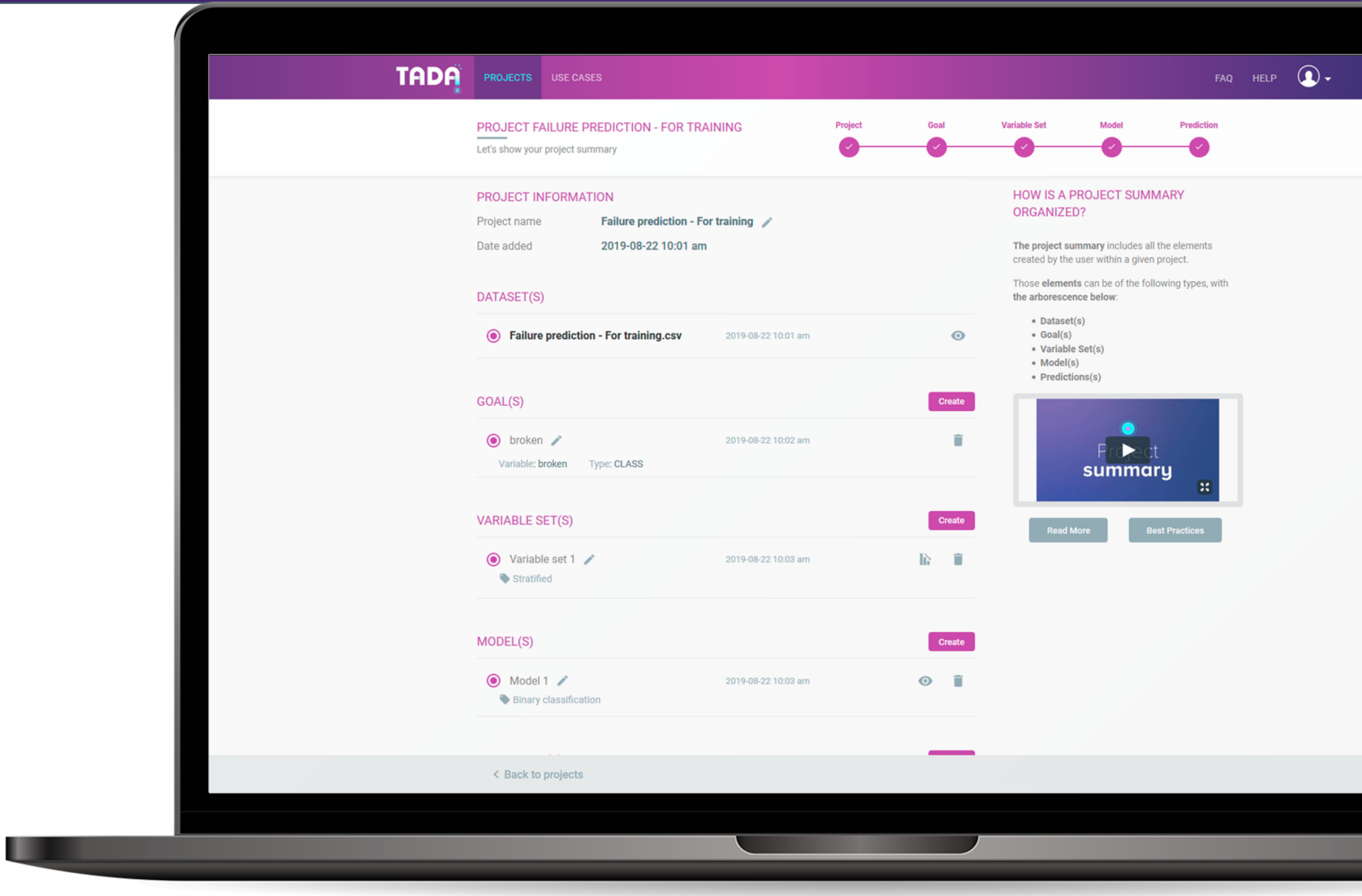
MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

TADA demo: model generation

TADA
enlighten your data

Let’s show how it works



PROJECT FAILURE PREDICTION - FOR TRAINING

Let's show your project summary

Project — Goal — Variable Set — Model — Prediction

PROJECT INFORMATION

Project name	Failure prediction - For training
Date added	2019-08-22 10:01 am

DATASET(S)

Failure prediction - For training.csv 2019-08-22 10:01 am

GOAL(S) Create

broken 2019-08-22 10:02 am
Variable: broken Type: CLASS

VARIABLE SET(S) Create

Variable set 1 2019-08-22 10:03 am
Stratified

MODEL(S) Create

Model 1 2019-08-22 10:03 am
Binary classification

< Back to projects

HOW IS A PROJECT SUMMARY ORGANIZED?

The project summary includes all the elements created by the user within a given project.

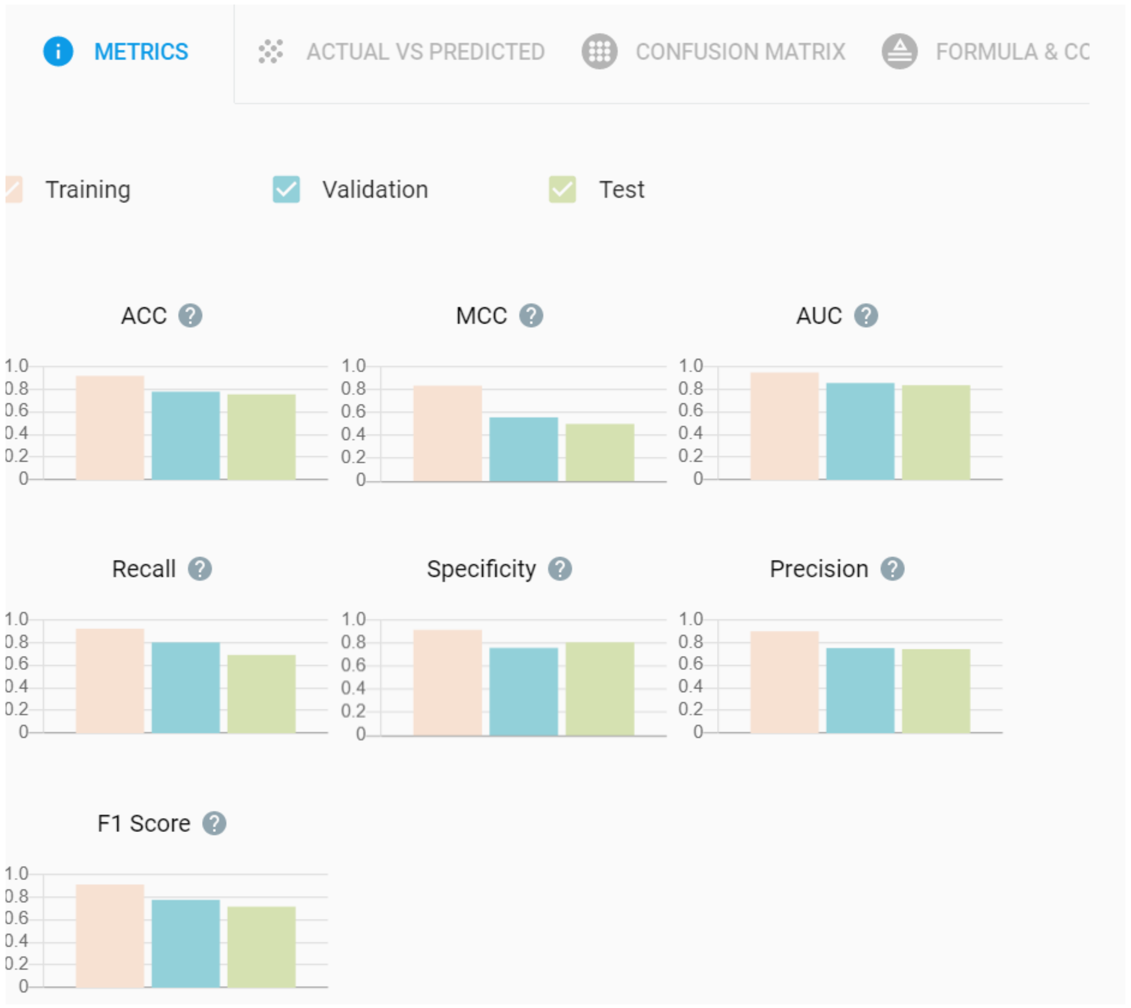
Those **elements** can be of the following types, with **the arborescence below**:

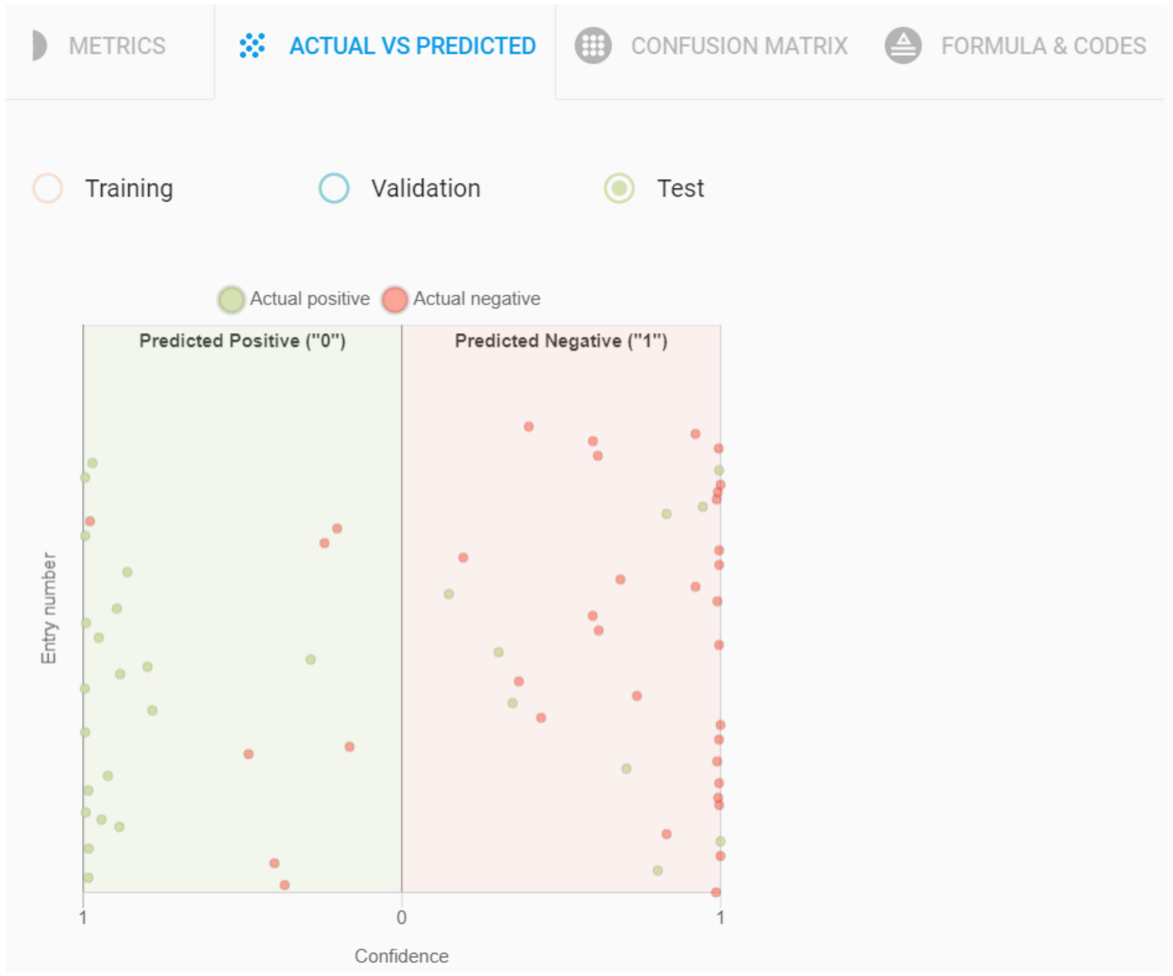
- Dataset(s)
- Goal(s)
- Variable Set(s)
- Model(s)
- Predictions(s)

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

TADA model generation: results (I)

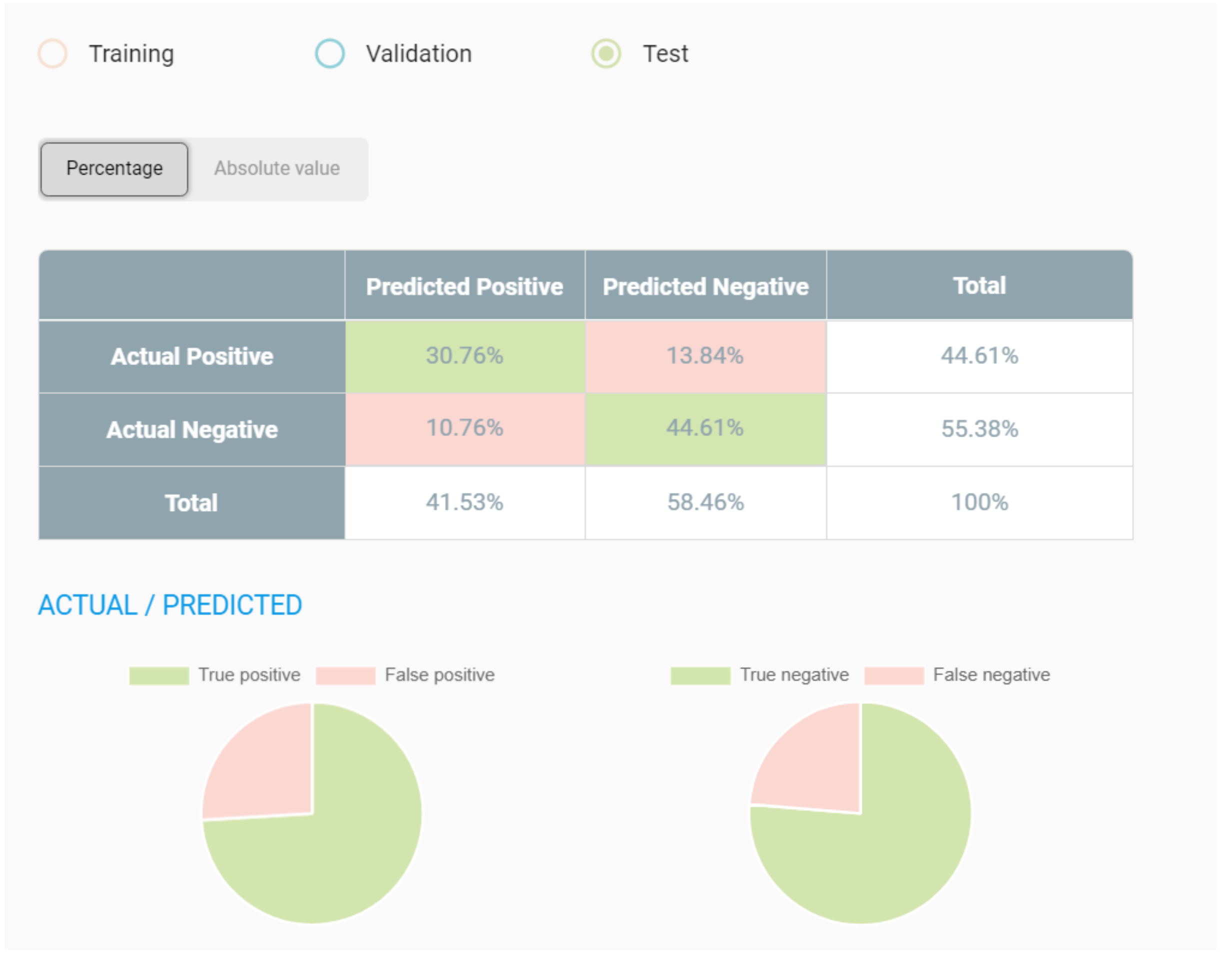




MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

TADA model generation: results (II)

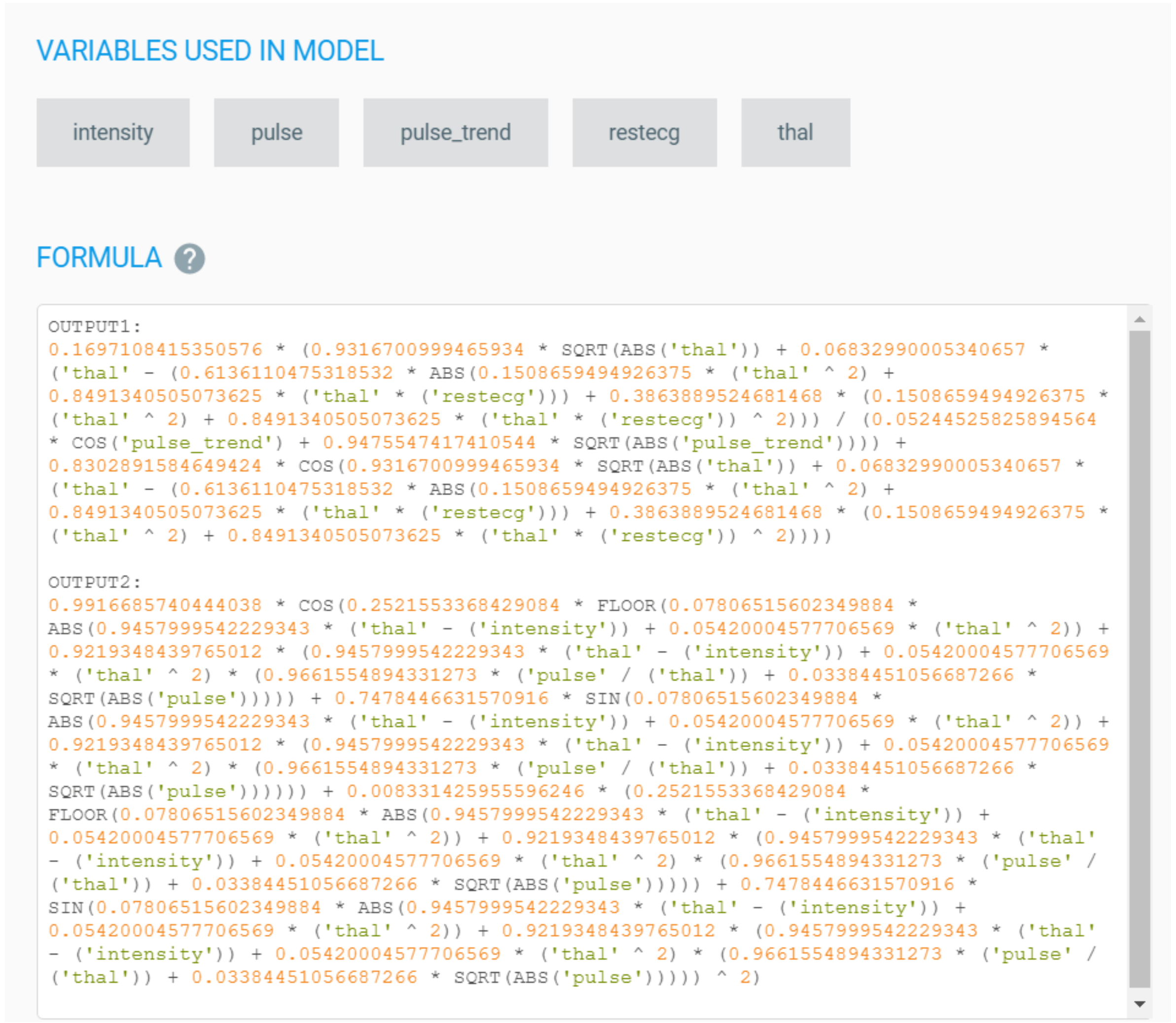


Training   Validation   [✓ Test]

[✓ Percentage]   Absolute value

	Predicted Positive	Predicted Negative	Total
Actual Positive	30.76%	13.84%	44.61%
Actual Negative	10.76%	44.61%	55.38%
Total	41.53%	58.46%	100%

ACTUAL / PREDICTED



VARIABLES USED IN MODEL

intensity   pulse   pulse_trend   restecg   thal

FORMULA

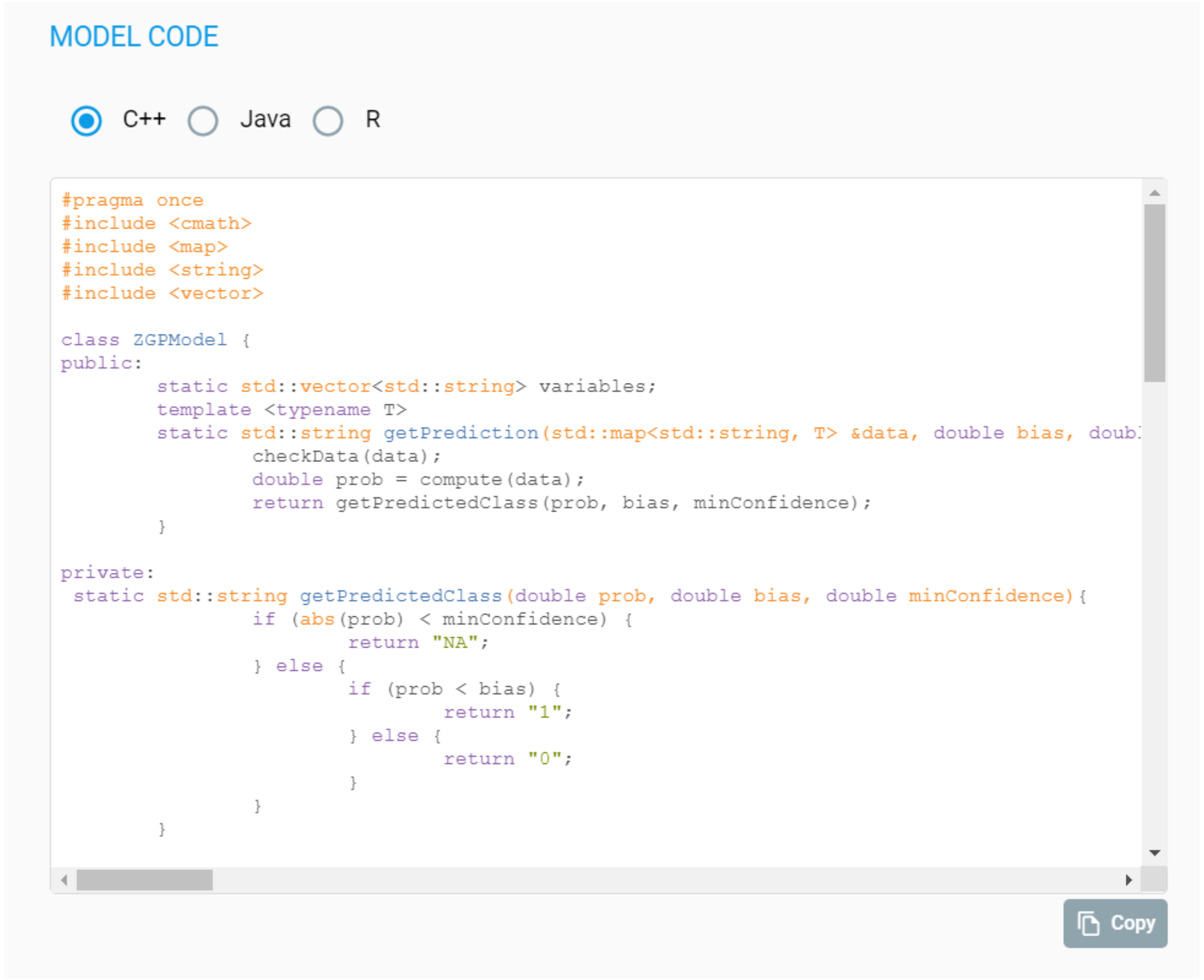
```
OUTPUT1:
0.1697108415350576 * (0.9316700999465934 * SQRT(ABS('thal')) + 0.06832990005340657 *
('thal' - (0.6136110475318532 * ABS(0.1508659494926375 * ('thal' ^ 2) +
0.849134050507362 5 * ('thal' * ('restecg'))) + 0.3863889524681468 * (0.1508659494926375 *
('thal' ^ 2) + 0.8491340505073625 * ('thal' * ('restecg')) ^ 2))) / (0.0524452582589456 4
* COS('pulse_trend') + 0.947554741741054 4 * SQRT(ABS('pulse_trend')))) +
0.8302891584649424 * COS(0.9316700999465934 * SQRT(ABS('thal')) + 0.06832990005340657 *
('thal' - (0.6136110475318532 * ABS(0.1508659494926375 * ('thal' ^ 2) +
0.8491340505073625 * ('thal' * ('restecg'))) + 0.3863889524681468 * (0.1508659494926375 *
('thal' ^ 2) + 0.8491340505073625 * ('thal' * ('restecg')) ^ 2))))

OUTPUT2:
0.9916685740444038 * COS(0.2521553368429084 * FLOOR(0.0780651560234988 4 *
ABS(0.9457999542229343 * ('thal' - ('intensity')) + 0.05420004577706569 * ('thal' ^ 2)) +
0.9219348439765012 * (0.9457999542229343 * ('thal' - ('intensity')) + 0.05420004577706569
* ('thal' ^ 2) * (0.9661554894331273 * ('pulse' / ('thal')) + 0.03384451056687266 *
SQRT(ABS('pulse'))))) + 0.7478446631570916 * SIN(0.07806515602349884 *
ABS(0.9457999542229343 * ('thal' - ('intensity')) + 0.05420004577706569 * ('thal' ^ 2)) +
0.9219348439765012 * (0.9457999542229343 * ('thal' - ('intensity')) + 0.05420004577706569
* ('thal' ^ 2) * (0.9661554894331273 * ('pulse' / ('thal')) + 0.03384451056687266 *
SQRT(ABS('pulse')))))) + 0.008331425955596246 * (0.2521553368429084 *
FLOOR(0.07806515602349884 * ABS(0.9457999542229343 * ('thal' - ('intensity')) +
0.05420004577706569 * ('thal' ^ 2)) + 0.9219348439765012 * (0.9457999542229343 * ('thal'
- ('intensity')) + 0.05420004577706569 * ('thal' ^ 2) * (0.9661554894331273 * ('pulse' /
('thal')) + 0.03384451056687266 * SQRT(ABS('pulse'))))) + 0.7478446631570916 *
SIN(0.07806515602349884 * ABS(0.9457999542229343 * ('thal' - ('intensity')) +
0.05420004577706569 * ('thal' ^ 2)) + 0.9219348439765012 * (0.9457999542229343 * ('thal'
- ('intensity')) + 0.05420004577706569 * ('thal' ^ 2) * (0.9661554894331273 * ('pulse' /
('thal')) + 0.03384451056687266 * SQRT(ABS('pulse'))))) ^ 2)
```

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

TADA model generation: results (II) and deployment



MODEL CODE

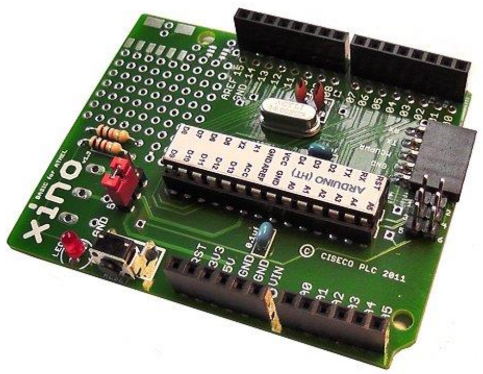
[✓ C++] Java R

```
#pragma once
#include <cmath>
#include <map>
#include <string>
#include <vector>

class ZGPModel {
public:
        static std::vector<std::string> variables;
        template <typename T>
        static std::string getPrediction(std::map<std::string, T> &data, double bias, doubl
                checkData(data);
                double prob = compute(data);
                return getPredictedClass(prob, bias, minConfidence);
        }

private:
 static std::string getPredictedClass(double prob, double bias, double minConfidence){
                if (abs(prob) < minConfidence) {
                        return "NA";
                } else {
                        if (prob < bias) {
                                return "1";
                        } else {
                                return "0";
                        }
                }
        }
```

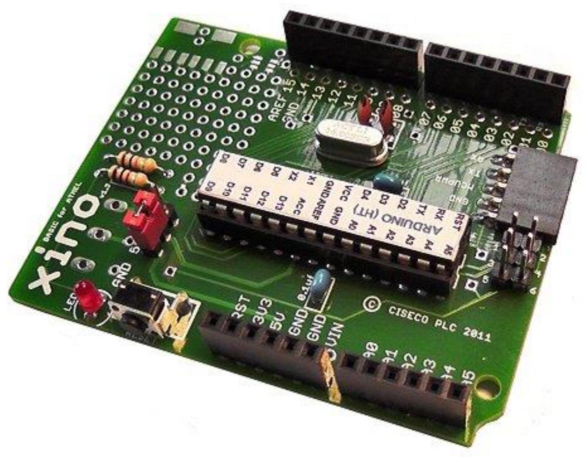
deploy



MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

TADA opens a new era of edge computing



TADA enables local decision making and added services for embedded systems - Efficient & Save

- TADA models (lower than 8 Kb) can be **embedded** into edge computers & microcontroller devices,
- No concern for data security, data are locally processed directly on the field where they are collected,
- Edge computing allows **lower latency** and reduces costs with a greater **reliability** than more traditional cloud computing approach using APIs.

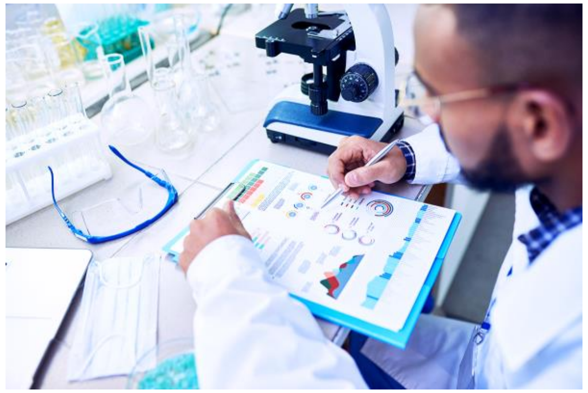
Make a decision locally on a vehicle, a machine, a plane, a satellite or any embedded system in milliseconds and complete safety

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

A wide variety of use cases for domain

HEALTHCARE



HOW TO PREDICT AND PREVENT ILLNESSES?

Using predictive models can save precious time to doctors in heart diseases prediction

INDUSTRY

HOW DO WE PREVENT FAILURES AND SET UP A PREDICTIVE MAINTENANCE SYSTEM?

Predicting failure before it ever happens

ENVIRONMENT

HOW DO WE PREDICT THE QUALITY OF OUR ENVIRONMENT AND IMPROVE IT?

Predicting the quality of our environment and improving it

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Our Partners & Eco-System

Inria

UNIVERSITÉ CÔTE D'AZUR

INCUBATEUR PACA-EST Innover, Entreprendre

TelecomValley Animateur Azuréen du Numérique

POLESCS

Cluster IA

MYDATAMODELS

THE SMALL DATA
PREDICTIVE MODELING
C O M P A N Y

Our References

THALES

Schneider Electric

Inserm
La science pour la santé
From science to health

cnrs

SANOFI

TEAM HENRI-FABRE
TECHNOLOGIES & EXPERTISE IN ADVANCED MANUFACTURING

Centre Hospitalier Universitaire de Nice

gemalto
a Thales company

INRA
SCIENCE & IMPACT

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Introducing Evolutionary Algorithms

Content

MYDATAMODELS

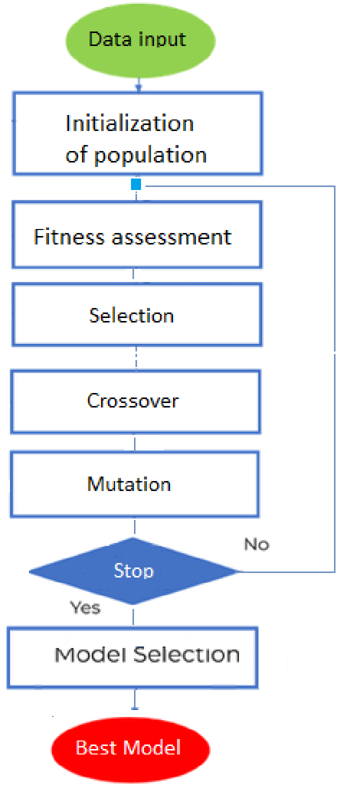
THE SMALL DATA PREDICTIVE MODELING COMPANY

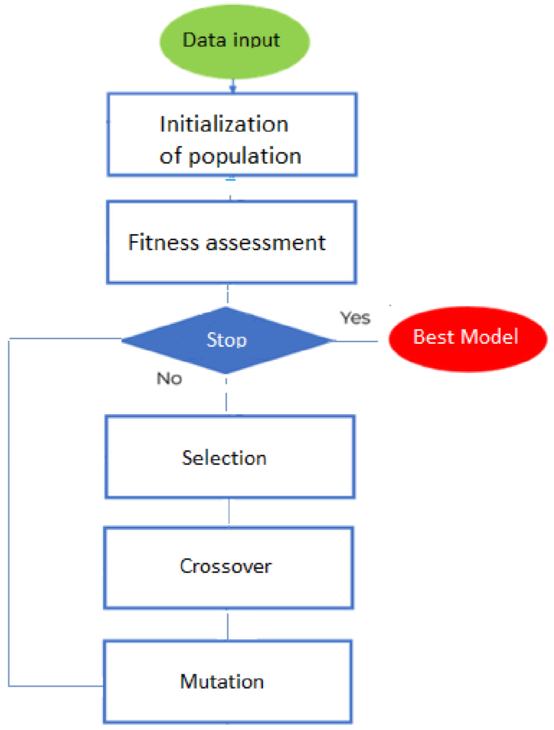
Introduction: Evolutionary Algorithms (EA) mimic the Darwinian "Survival of the Fittest"(*)

A 'Genetic Algorithm' (GA) is a search-based optimization technique used to find optimal solutions to 'difficult' problems, otherwise long to solve[1]. How?

1. A "pool" or a "population" of possible solutions to a given problem, is generated randomly
2. A fitness function is established for each
3. A parent selection mechanisms among the population is performed to allow recombination ("crossover") in order to produce new children ("the offspring")
4. Each individual (or candidate solution) is assigned a fitness function. The fitter individuals are given a higher chance to mate and generate more "fitter" individuals…("recombination or crossover")
5. Mutation is a random variation imposed to the individuals which contributes to additional 'variability'.

The process is repeated over various generations (often the stop criterion is a fixed number of "generations").





(*)For example, Genetic Algorithms, E D Goodman, Michigan State Univ: https://slideplayer.com/slide/5339767/
A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, 2nd ed. Springer 2015
https://blog.overops.com/how-to-solve-tough-problems-using-genetic-algorithms/
https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_introduction.htm

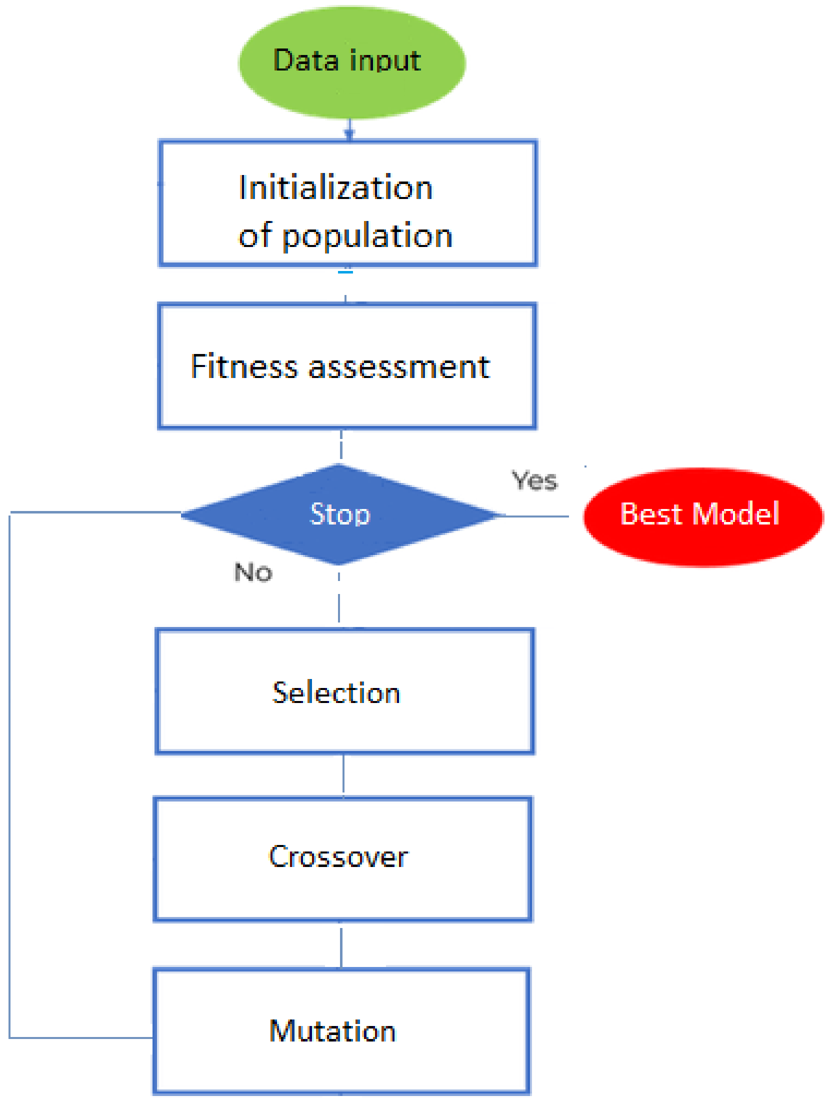
MYDATAMODELS | THE SMALL DATA PREDICTIVE MODELING COMPANY

Introduction: Evolutionary Algorithms (EA) mimic the Darwinian "Survival of the Fittest"(*)

A 'Genetic Algorithm' (GA) is a search-based optimization technique used to find optimal solutions to 'difficult' problems, otherwise long to solve[1]. How?

1. A "pool" or a "population" of possible solutions to a given problem, is generated randomly
2. A fitness function is established for each
3. A parent selection mechanisms among the population is performed to allow recombination ("crossover") in order to produce new children ("the offspring")
4. Each individual (or candidate solution) is assigned a fitness function. The fitter individuals are given a higher chance to mate and generate more "fitter" individuals…("recombination or crossover")
5. Mutation is a random variation imposed to the individuals which contributes to additional 'variability'.

The process is repeated over various generations (often the stop criterion is a fixed number of "generations").



THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

‘Genetic Algorithms’: nomenclature

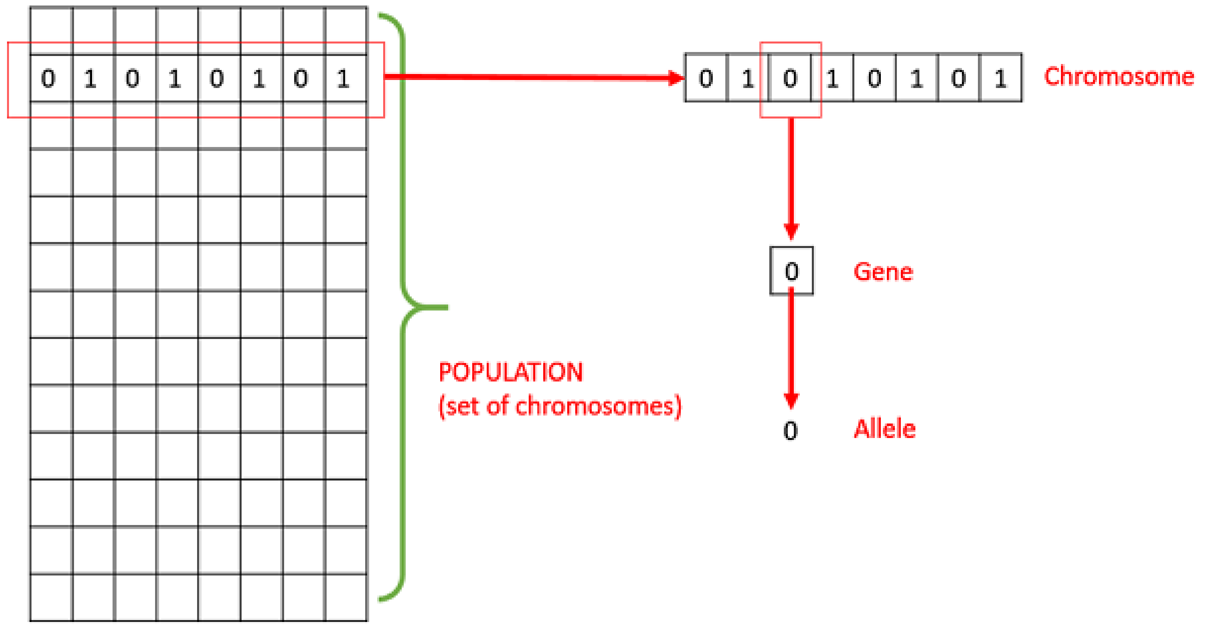
Population – subset of all the possible (encoded) solutions – individuals - to the given problem, each characterized by a set of parameters (variables) known as *genes*

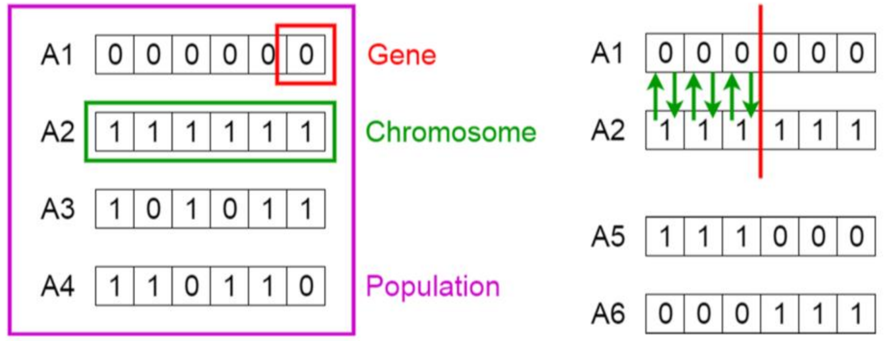
Chromosomes – A chromosome is one such solution to the given problem (individual or element in our nomenclature

Gene – A gene is one element position of a chromosome: here we have GA with encoded string as chromosome, thus one gene is one bit - literally

Allele – It is the value a gene takes for a particular chromosome.

Fitness function that gets a candidate solution to the problem as input and produces as output how “good” the solution to the problem is.





https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_fundamentals.htm

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

‘Genetic Algorithms’ ∈ Evolutionary Computing

In the frame of Optimization, modelling and in Search problems

Evolutionary algorithms (‘evolutionary computing/computation’ / ‘Optimization algorithms’ since the 1950s - Mitchell 1998)

exist in four major approaches, differing mostly in *selection methods*, *representation schemes* (‘genetic representation’ of the individual), *reproduction operators*:

- Genetic Algorithms (GA), (Holland, 1975)
- Genetic Programming (GP), (Koza, 1992, 1994),
- Evolutionary Strategies (ES), (Rechenberg, 1973),
- Evolutionary Programming (EP), (Fogel et al., 1966)

>40 years old!

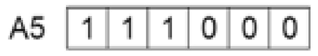
Hence, in this presentation ‘Genetic Algorithms’ is ~ok when used generically to explain the overall filed

→ but it’s bad practice, because it is <u>incorrect</u>

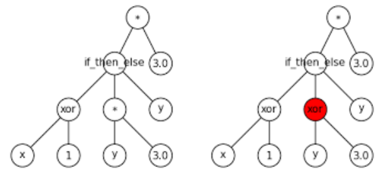
Stefan Droste, Thomas Jansen, Ingo Wegener, On the analysis of the (1 + 1) evolutionary algorithm , Theoretical Computer Science 276 (2002) 51–81
A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, Second Edition, Springer, ISBN 978-3-662-44873-1

'Genetic Algorithms' ∈ Evolutionary Computing

Genetic Algorithms (GA), (Holland, 1975)
Representation: **fixed-length bit string**, Each position = particular feature, string "evaluated as a collection of features with little or no interactions".
Genetic operators: Crossover, no external info introduced; Mutation, external (random) information introduced



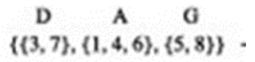
Genetic Programming (GP), (Koza, 1992, 1994)
Representation: **variable-sized tree of functions** and values. Leaf = label from available label set entire tree corresponds to a single function.
reproduction operators tailored to tree representation. Most common operators: subtree crossover, entire subtree swapped between two parents. In a standard genetic program, all values and functions return the same type



Evolutionary Strategies (ES), (Rechenberg, 1973)
Representation: **fixed-length real-valued vector**. Like bit strings of GA, each position = feature of the individual.
Reproduction operator es is Gaussian mutation: random value from a Gaussian distribution added to each element of an individual's vector to create new offspring. Another operator: intermediate recombination, where vectors of two parents are averaged element by element, to form a new offspring

D A G
{{3, 7}, {1, 4, 6}, {5, 8}}

Evolutionary Programming (EP), (Fogel et al., 1966)
Representation: tailored to the problem domain, **fixed-length real-valued vector**, no exchange between individuals in the population is made. Thus, only mutation used. For real-valued vector representations, evolutionary programming is very similar to evolutionary strategies but *without* recombination.



THE SMALL DATA PREDICTIVE MODELING COMPANY
MYDATAMODELS

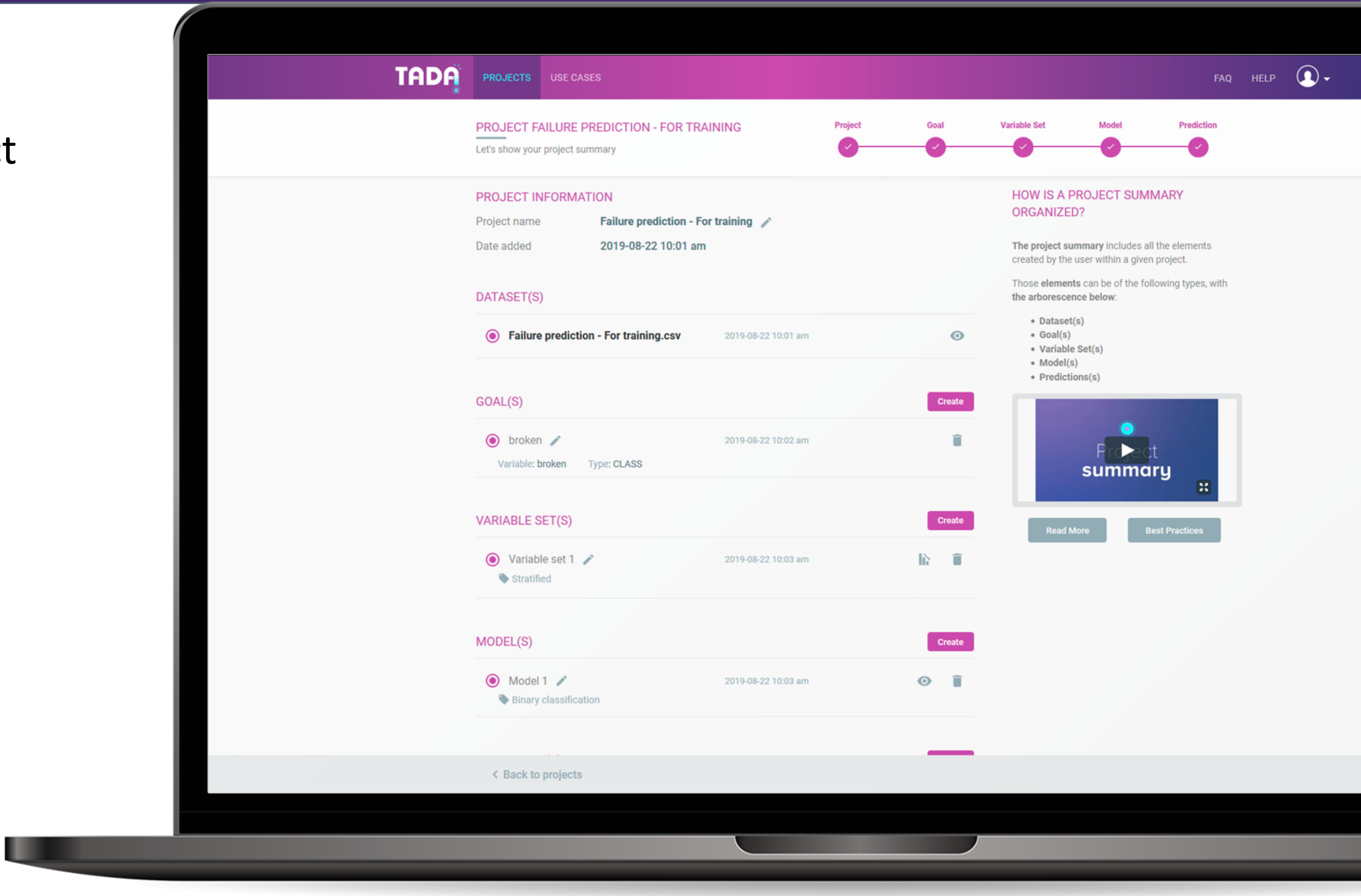
Evolutionary Algorithms within MDM product

These principles are implemented in **TADA:** our product

within its ZGP Engine: 'Zoetrope' Genetic Program, based on *Genetic Algorithms*(*)

- generate and run *automated* predictive models on client's *own data*
- Easy to use without programming or *background in machine learning*.
- performs well on *Small Data*.

As opposed to Big Data, '**Small' Data** involve 'small' samples , ~ hundreds or fewer' observations…



PROJECT FAILURE PREDICTION - FOR TRAINING
Let's show your project summary

PROJECT INFORMATION

Project name	Failure prediction - For training
Date added	2019-08-22 10:01 am

DATASET(S)

Failure prediction - For training.csv 2019-08-22 10:01 am

GOAL(S)

broken 2019-08-22 10:02 am
Variable: broken Type: CLASS

VARIABLE SET(S)

Variable set 1 2019-08-22 10:03 am
Stratified

MODEL(S)

Model 1 2019-08-22 10:03 am
Binary classification

HOW IS A PROJECT SUMMARY ORGANIZED?

The project summary includes all the elements created by the user within a given project.

Those **elements** can be of the following types, with **the arborescence below**:

- Dataset(s)
- Goal(s)
- Variable Set(s)
- Model(s)
- Predictions(s)

(*) 'Zoe…' what? One of coming slides

Evolutionary Algorithms: foundations

'Zoe… Zoetrope' ? Wikipedia helps us:

(Ed.) The zoetrope is a cylinder with vertical cuts in the sides.

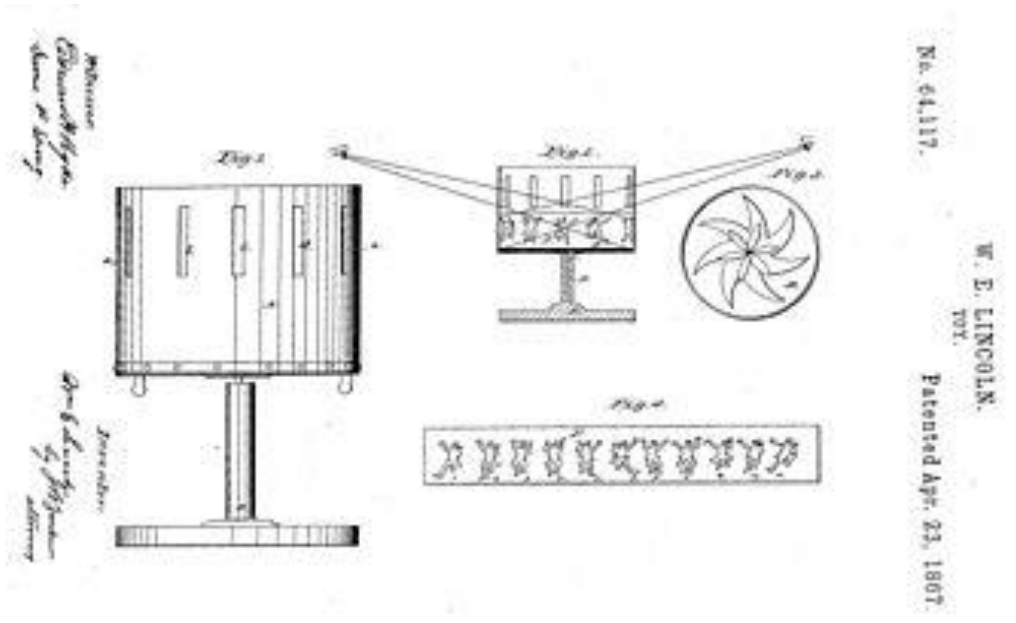
On the inner surface of the cylinder is a band with images from a set of sequenced pictures.

As the cylinder spins, the user looks through the cuts at the pictures across.

The scanning of the slits keeps the pictures from simply blurring together, and the user sees a rapid succession of images, producing the ***illusion of motion.***
Watch this: *https://www.youtube.com/watch?v=5_8fX-N3Ji4*

. . .

Worth mentioning 1868 **James Clerk Maxwell** improved version, with cuts changed into lenses with focal length = diameter of the circle, so to get a sharp and stable image to the center…



W.E. Lincoln's U.S. Patent No. 64,117 of Apr. 23, 1867

Evolutionary Algorithms: fitness

How to define a **fitness?**

Fitness might coincide with an objective function (but not necessarily…)

Example

Find values for a set of variables satisfying a given constraint; like, given x, y and z, find the best set of x, y, z values such that their total is equal to a value u.

$$x + y + z = u$$ **Objective**

Task: we need to reduce the difference $|x + y + z - u|$ as much as possible so to approach zero(*).
Therefore, we may consider as fitness the following function:

$$f = 1 / |x + y + z - u|$$ **Fitness**

(*) nearly, else the fraction diverges

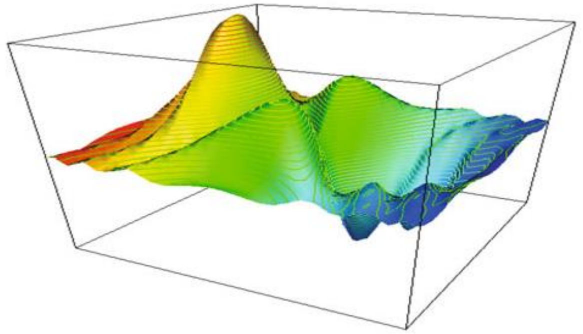
MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Evolutionary Algorithms: fitness

Population = # of individuals, whose reproductive success depends on how each adapt to environment relative to the rest.
The more successful reproduce, and
occasional mutations give rise to new individuals to be tested.
As time passes, there is a change in the population (population is the unit of evolution)

Darwinian evolution to adaptive landscape[1]: the height belongs to fitness: high altitude →high fitness.
The other two (or more) dimensions → 'biological' traits


Adaptive landscape with two traits[1]

The *xy*-plane holds all possible trait combinations, and the *z* values their fitness.
Each peak is a range of successful combinations; the troughs: less-fit combinations
Risk exists of getting stuck in 'local optimum'→ mutation might help

[1]A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, Second Edition, Springer, ISBN 978-3-662-44873-1

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

Genetic operators

Selection

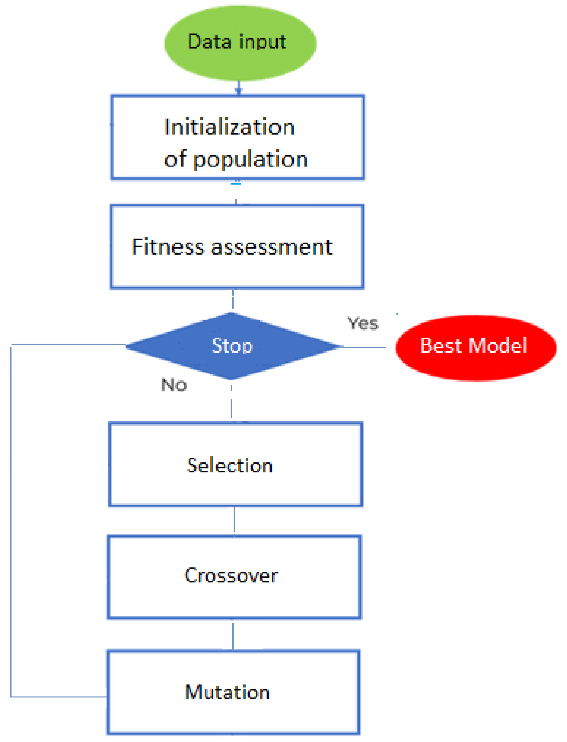
(Parent) Selection: selecting parents which mate and recombine to create off-springs for the next generation. Crucial to the convergence rate of the GA

Crossover

(also: “recombination”): analogous to reproduction: More than one parent is selected to produce more than one offspring using parents “genetic material”

Mutation

add randomness to maintain and introduce diversity in the genetic population



Hans-Georg Beyera, Hans-Paul Schwefela, Ingo Wegener, How to analyse evolutionary algorithms,
Theoretical Computer Science 287 (2002) 101–130
https://reader.elsevier.com/reader/sd/pii/S0304397502001378?token=581643E4DB2498BE74F1CAA4289F1587DC587E6FD0305E4BE5B901D855ECE419A7D023D5BEAD425B8C2AED3B47CC325E#cite.79
https://data-flair.training/blogs/python-genetic-algorithms-ai/
https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_parent_selection.htm

Genetic operators

Selection

Initialization of the population of chromosomes or Genotypes or ‘binary strings’ (for GA)
Random generation of elements
Evaluation – fitness (often identified, in general distinct functions) selection (“Parent selection”)

Via Fitness proportionate selection
probability to become a parent proportional to its fitness $\Rightarrow$

- Roulette Wheel Selection
- Stochastic Universal sampling Selection
- Tournament Selection
- Rank Selection (use fitness only to rank and select from rank)
- Random Selection

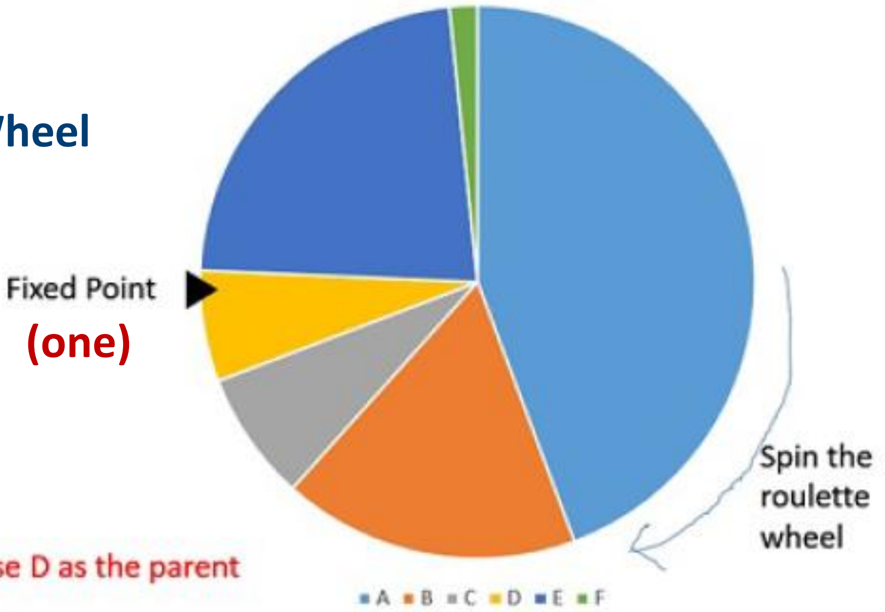
THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

Genetic operators: (parents) selection

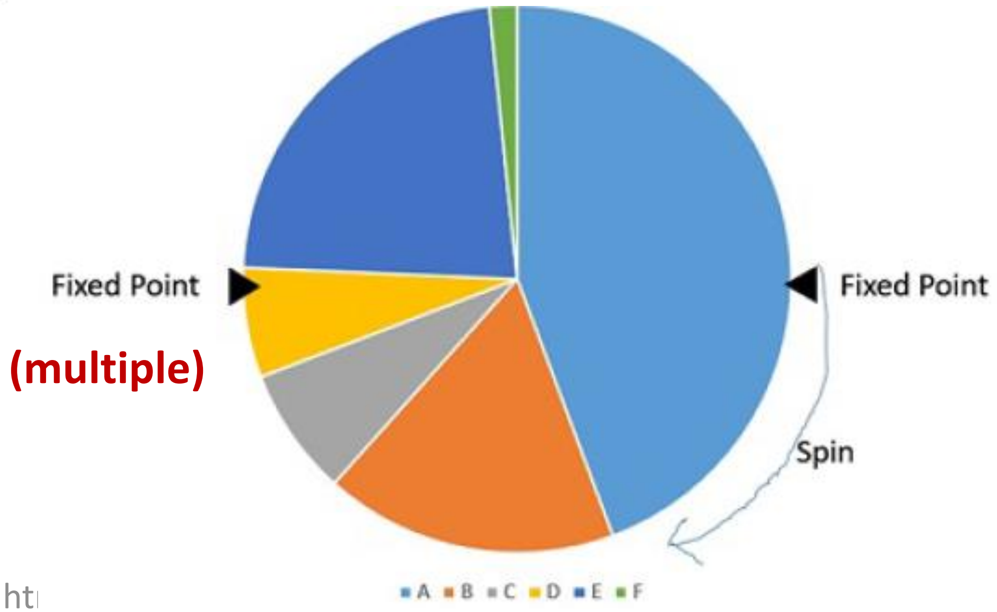
Selection

- **Roulette Wheel Selection**



Chromosome	Fitness Value
A	8.2
B	3.2
C	1.4
D	1.2
E	4.2
F	0.3

- **Stochastic Universal sampling Selection**



Chromosome	Fitness Value
A	8.2
B	3.2
C	1.4
D	1.2
E	4.2
F	0.3

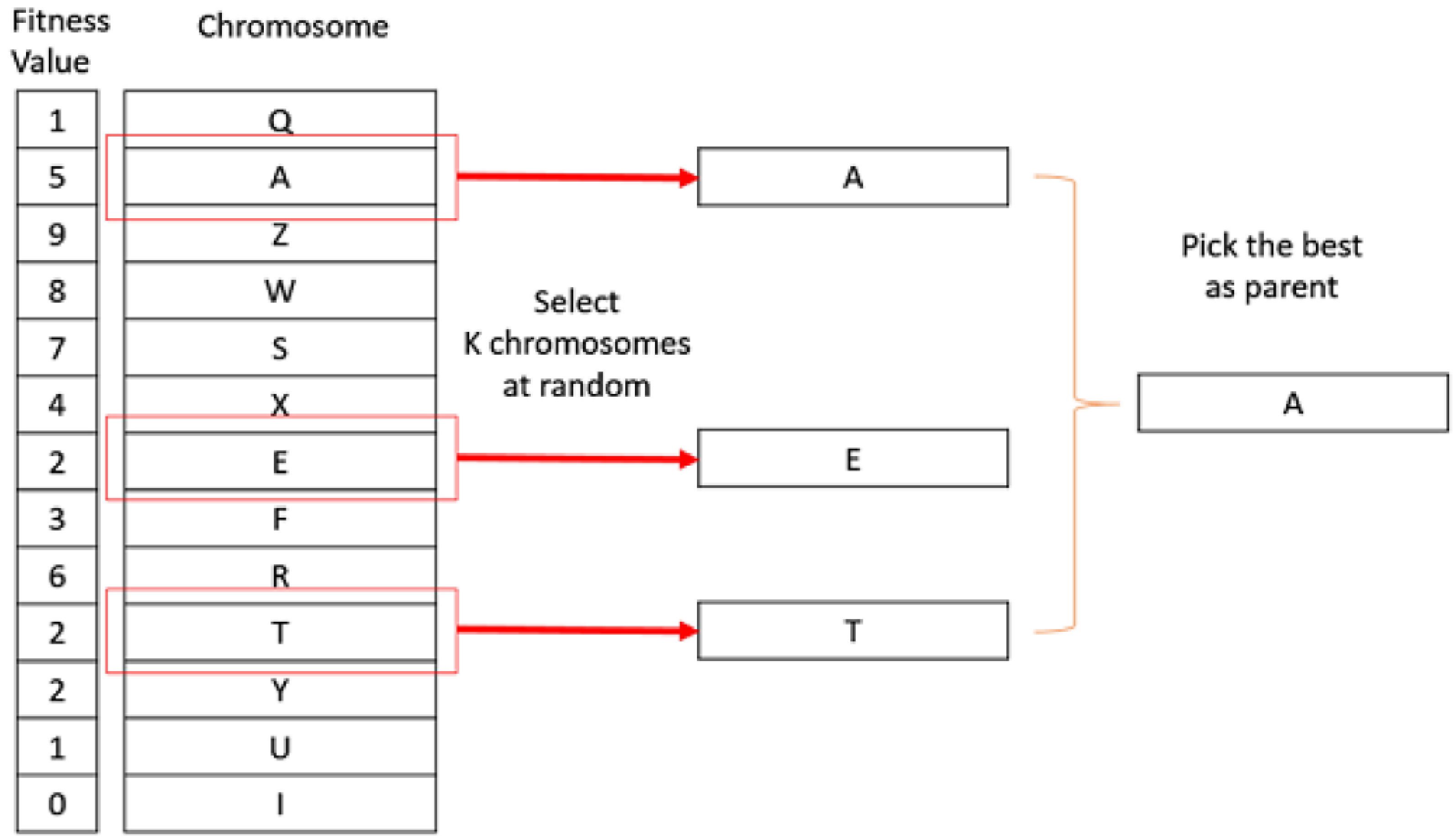
Genetic operators: (parents) selection

Selection

- **Tournament Selection**

Take K individuals at random and select the best as the parent

Repeat for each parent (can select several, depends on chosen strategy



Genetic operators: (parents) selection

Selection

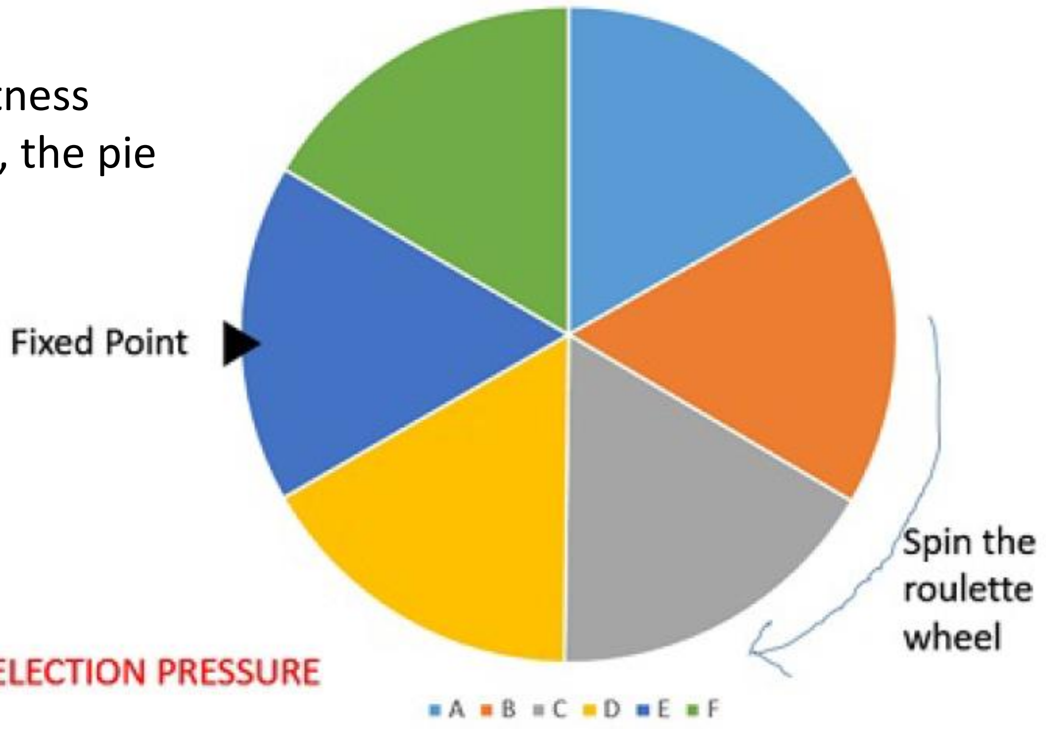
Rank selection

When individuals in the population have very close fitness values, ~ same probability of being chosen as parents, the pie slices are the same or

→ lower 'selection pressure'

→ choice might be non optimal.

So, remove fitness value but use it to rank while selecting a parent (like dilating differences)
→ rank of each individual and not the fitness



Chromosome	Fitness Value
A	8.1
B	8.0
C	8.05
D	7.95
E	8.02
F	7.99

Random selection

Randomly select parents from the existing population…

MYDATAMODELS

THE SMALL DATA PREDICTIVE MODELING COMPANY

Genetic operators: crossover

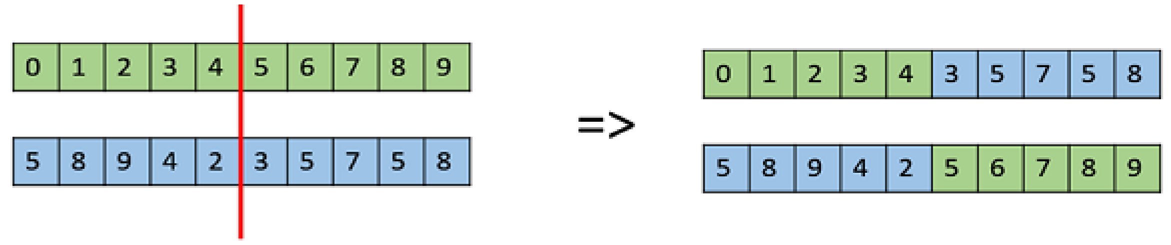
Crossover

(also “recombination”): > 1(one) parent selected to yield >1 offspring, using parents “genetic material”

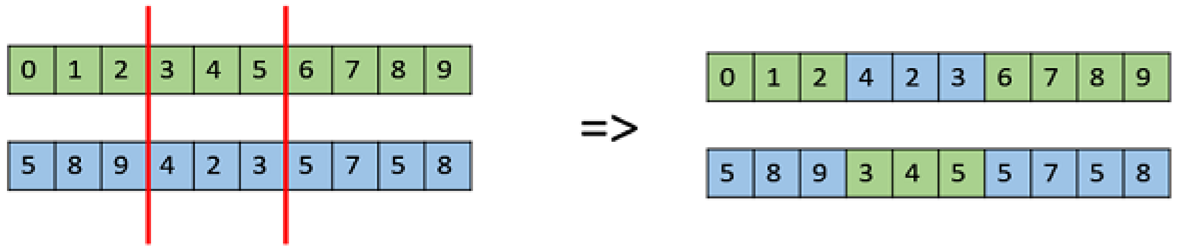
Popular crossover Operators (I)

(random crossover point)

One Point

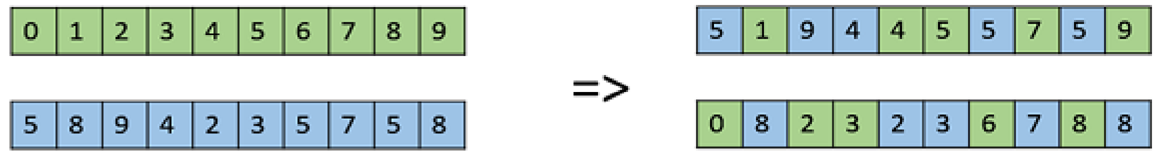


Multi Point



flip a coin for each chromosome to decide whether or not it’ll be included in the off-spring

Uniform Crossover



https://www.tutorialspoint.com/genetic_algorithms/

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

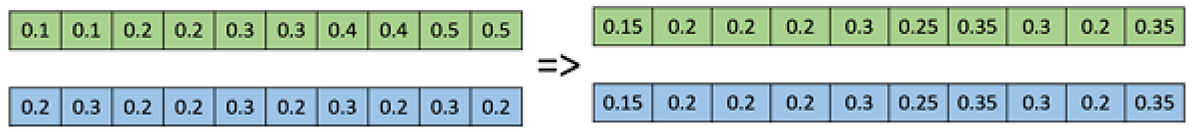
Genetic operators: crossover

Crossover

Popular operators (I)

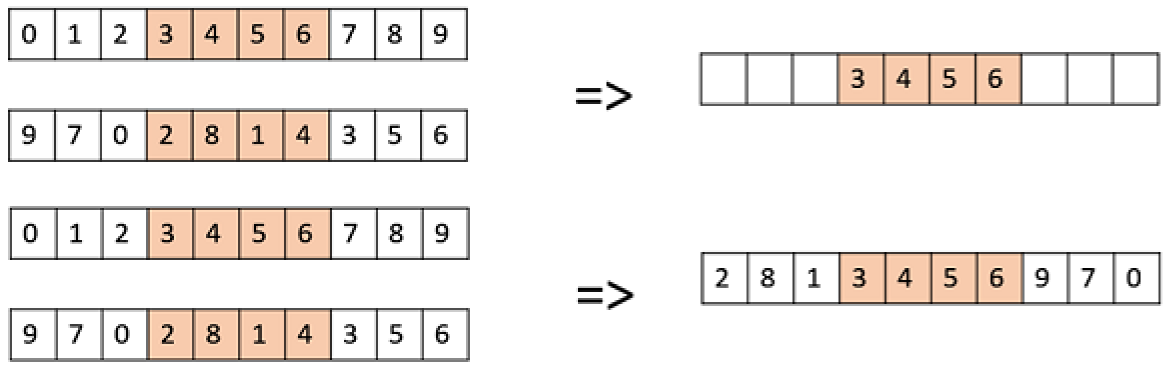
Whole Arithmetic Recombination

weighted average of the two parents



Davis' Order Crossover (OX1)

- Create two random crossover points in the parent and copy the segment between them from the first parent to the first offspring
- starting from the second crossover point in the second parent, copy the remaining unused numbers from the second parent to the first child, wrapping around the list.
- Repeat for the second child with the parent's role reversed.



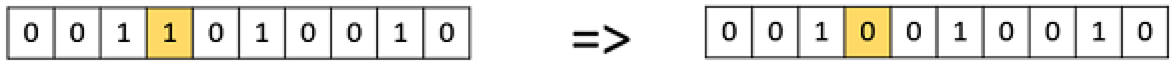
Many others as variations: Partially Mapped Crossover (PMX), Order based crossover (OX2), Shuffle Crossover, ….

Genetic operators: mutation

Mutation (genetic algorithm case)

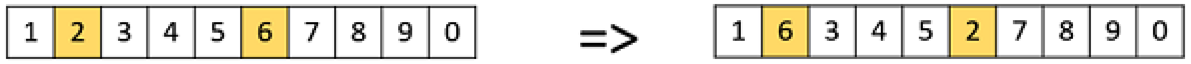
Add randomness to maintain and introduce diversity in the genetic population
Popular (generic) mutation operators

Bit Flip Mutation

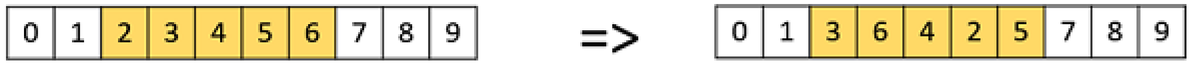


Random Resetting: extension of bit flip for integer representation: here, a random value from the set of permissible values is assigned to a randomly chosen gene

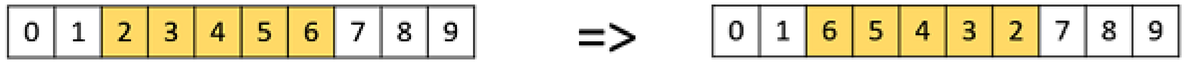
Swap Mutation



Scramble Mutation



Inversion Mutation



Genetic Algorithms, applications: when?

When is / are EP / GA better?

For difficult problems, for example they are
more indicated than neural networks, when the *solution space is non-differentiable*
*(*Problem with discontinuous piecewise linear cost function[*])

Ex. neural networks use gradient descent to learn from backpropagation
but computation of the gradient is based on derivatives (which require continuous and differentiable space).

In EP/GA this is not needed: rather than shifting ‘continuously’ among solutions, one
May jump from one solution to the next

[*] S Sheng, Xu Xiaofei, Genetic Algorithm for the Transportation Problem with Discontinuous Piecewise Linear Cost Function, IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.7A, July 2006
https://www.researchgate.net/publication/254308802_Genetic_Algorithm_for_the_Transportation_Problem_with_Discontinuous_Piecewise_Linear_Cost_Function

Genetic Algorithms , applications: which ones?

Where are EP / GA used?

In design:

MHD nozzles (Klockgether, J. & Schwefel, Hans-Paul. (1970). TWO-PHASE NOZZLE AND HOLLOW CORE JET EXPERIMENTS. pp 141-8 of Engineering Aspects of Magnetohydrodynamics. /Elliott, D. G. (ed.). University, Miss. Univ. of Mississippi (1970).!!

Optical lenses
Kaspar Hoschel, Vasudevan Lakshminarayanan, Genetic algorithms for lens design: a review, J Opt, March 2019, 48(1):134–144, https://doi.org/10.1007/s12596-018-0497-3

In planning/scheduling:
Air traffic
Dynamic airspace configuration (Marina Sergeeva, Daniel Delahaye, Catherine Mancel, Andrija Vidosavljevic, Dynamic airspace configuration by genetic algorithm, journal of traffic and transportation engineering (english edition) 2017 ; 4 (3) : 300-314)

Robot path planning (Er. Waghoo Parvez, Er. Sonal Dhar, Path Planning Optimization Using Genetic Algorithm–A Literature Review, International Journal of Computational Engineering Research, Vol, 3(4), 2013)

In manufacturing:
Additive manufacturing (Torbjørn Schjelderup Leirmo, Kristian Martinsen, Evolutionary algorithms in additive manufacturing systems: Discussion of future prospects, 52nd CIRP Conference on Manufacturing Systems, Procedia CIRP 81 (2019) 671–676

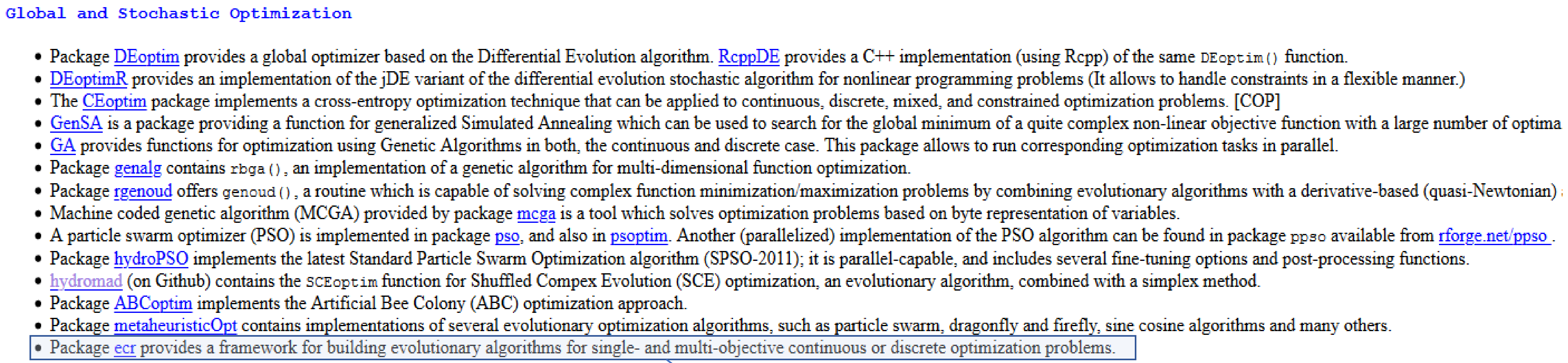
THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

Genetic Algorithms: open source frameworks

R + RStudio

Non-exhaustive list (even under the Optimization 'View': RFreak, GA, rgp…) mostly mono-objective



Global and Stochastic Optimization

- Package DEoptim provides a global optimizer based on the Differential Evolution algorithm. RcppDE provides a C++ implementation (using Rcpp) of the same `DEoptim()` function.
- DEoptimR provides an implementation of the jDE variant of the differential evolution stochastic algorithm for nonlinear programming problems (It allows to handle constraints in a flexible manner.)
- The CEoptim package implements a cross-entropy optimization technique that can be applied to continuous, discrete, mixed, and constrained optimization problems. [COP]
- GenSA is a package providing a function for generalized Simulated Annealing which can be used to search for the global minimum of a quite complex non-linear objective function with a large number of optima
- GA provides functions for optimization using Genetic Algorithms in both, the continuous and discrete case. This package allows to run corresponding optimization tasks in parallel.
- Package genalg contains `rbga()`, an implementation of a genetic algorithm for multi-dimensional function optimization.
- Package rgenoud offers `genoud()`, a routine which is capable of solving complex function minimization/maximization problems by combining evolutionary algorithms with a derivative-based (quasi-Newtonian)
- Machine coded genetic algorithm (MCGA) provided by package mcga is a tool which solves optimization problems based on byte representation of variables.
- A particle swarm optimizer (PSO) is implemented in package pso, and also in psoptim. Another (parallelized) implementation of the PSO algorithm can be found in package `ppso` available from rforge.net/ppso .
- Package hydroPSO implements the latest Standard Particle Swarm Optimization algorithm (SPSO-2011); it is parallel-capable, and includes several fine-tuning options and post-processing functions.
- hydromad (on Github) contains the `SCEoptim` function for Shuffled Compex Evolution (SCE) optimization, an evolutionary algorithm, combined with a simplex method.
- Package ABCoptim implements the Artificial Bee Colony (ABC) optimization approach.
- Package metaheuristicOpt contains implementations of several evolutionary optimization algorithms, such as particle swarm, dragonfly and firefly, sine cosine algorithms and many others.
- Package ecr provides a framework for building evolutionary algorithms for single- and multi-objective continuous or discrete optimization problems.

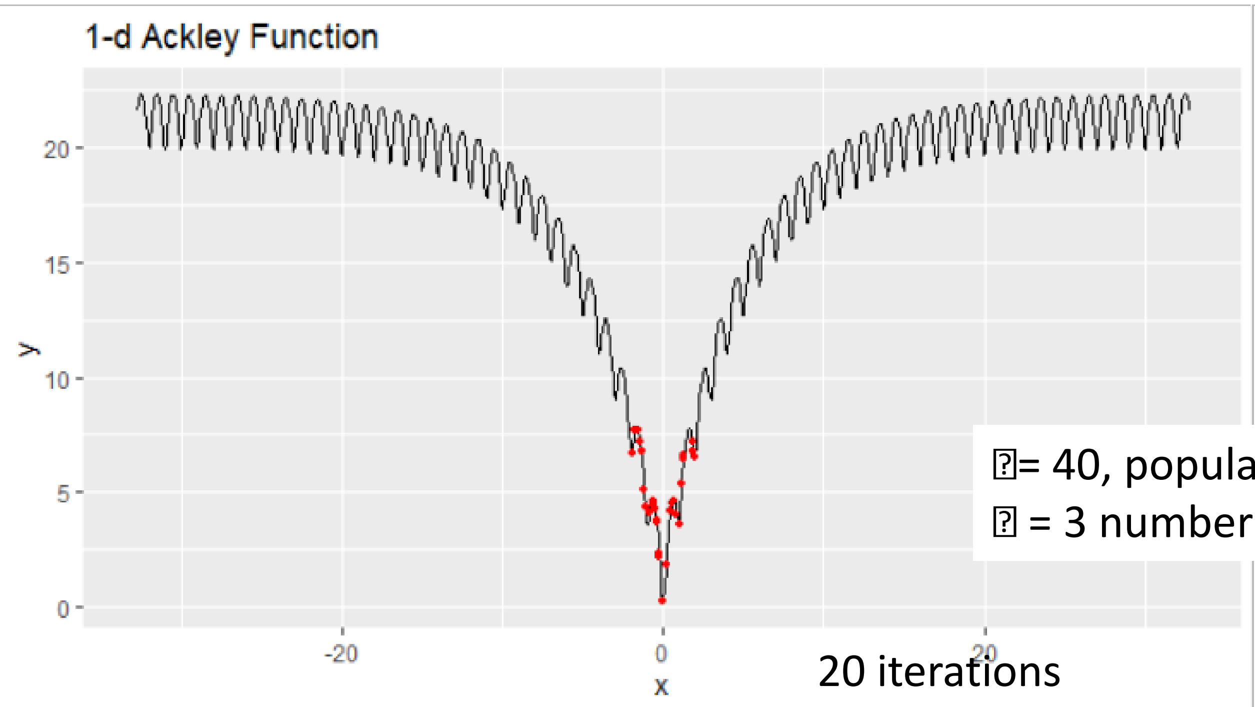
"inspired by the 'awesome Evolutionary Computation (EC) framework' DEAP for Python" (*cit.*)

https://cran.r-project.org/web/views/Optimization.html

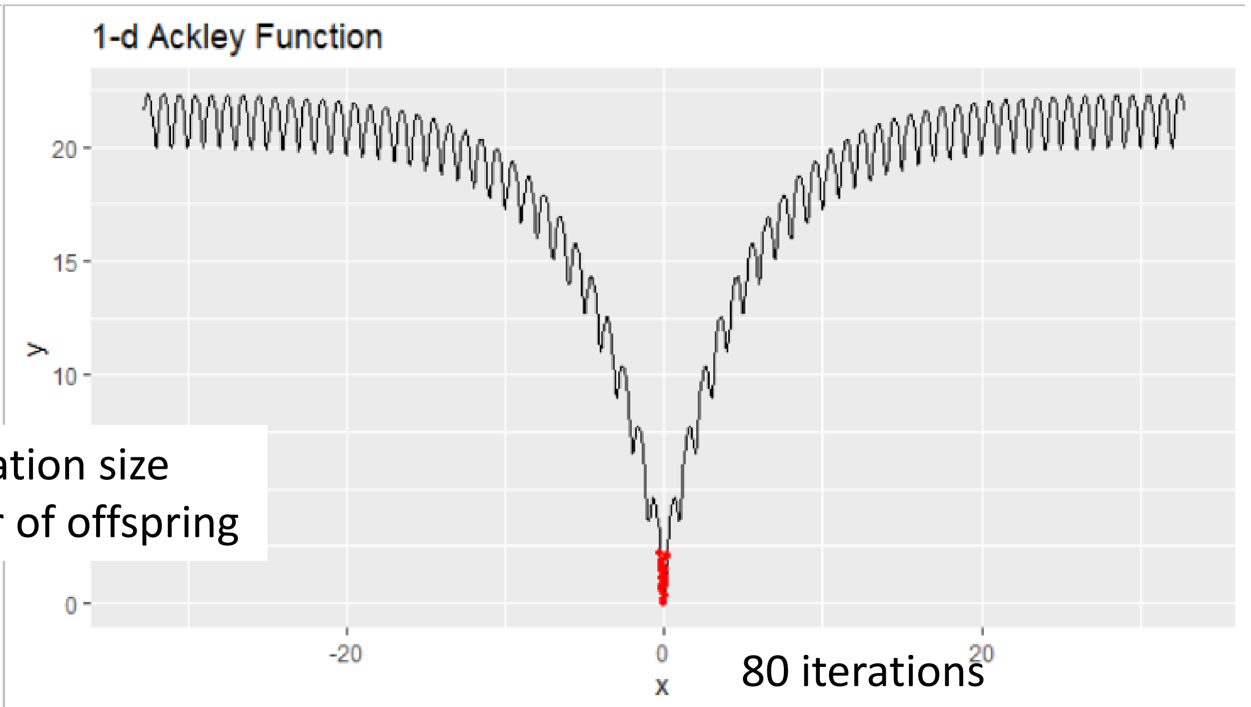
Genetic Algorithms: open source frameworks

Simple example od use of **ecr** library (@white box")

We run the ecr example with several iterations to show the problem of **too early stop**, example 20 iterations and 80 iterations with a (μ+λ) strategy, with *mutation* only.



20 iterations



80 iterations

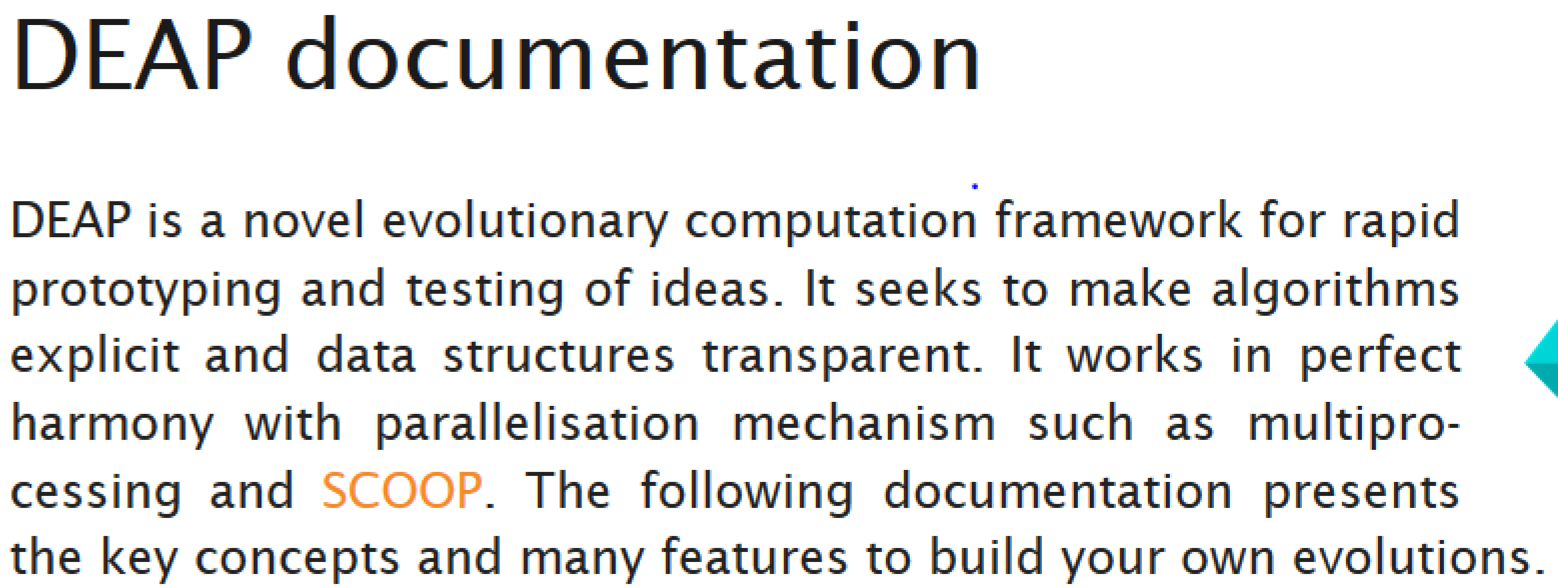
⍰= 40, population size
⍰ = 3 number of offspring

(μ+λ) strategy: the best survive to the next generation;(μ, λ) strategy: only *child individuals* survive to the next generation

The best EA to study mutation in isolation is the (1+1) EA

Genetic Algorithms: ...frameworks - DEAP

python



DEAP documentation

DEAP is a novel evolutionary computation framework for rapid prototyping and testing of ideas. It seeks to make algorithms explicit and data structures transparent. It works in perfect harmony with parallelisation mechanism such as multiprocessing and SCOOP. The following documentation presents the key concepts and many features to build your own evolutions.

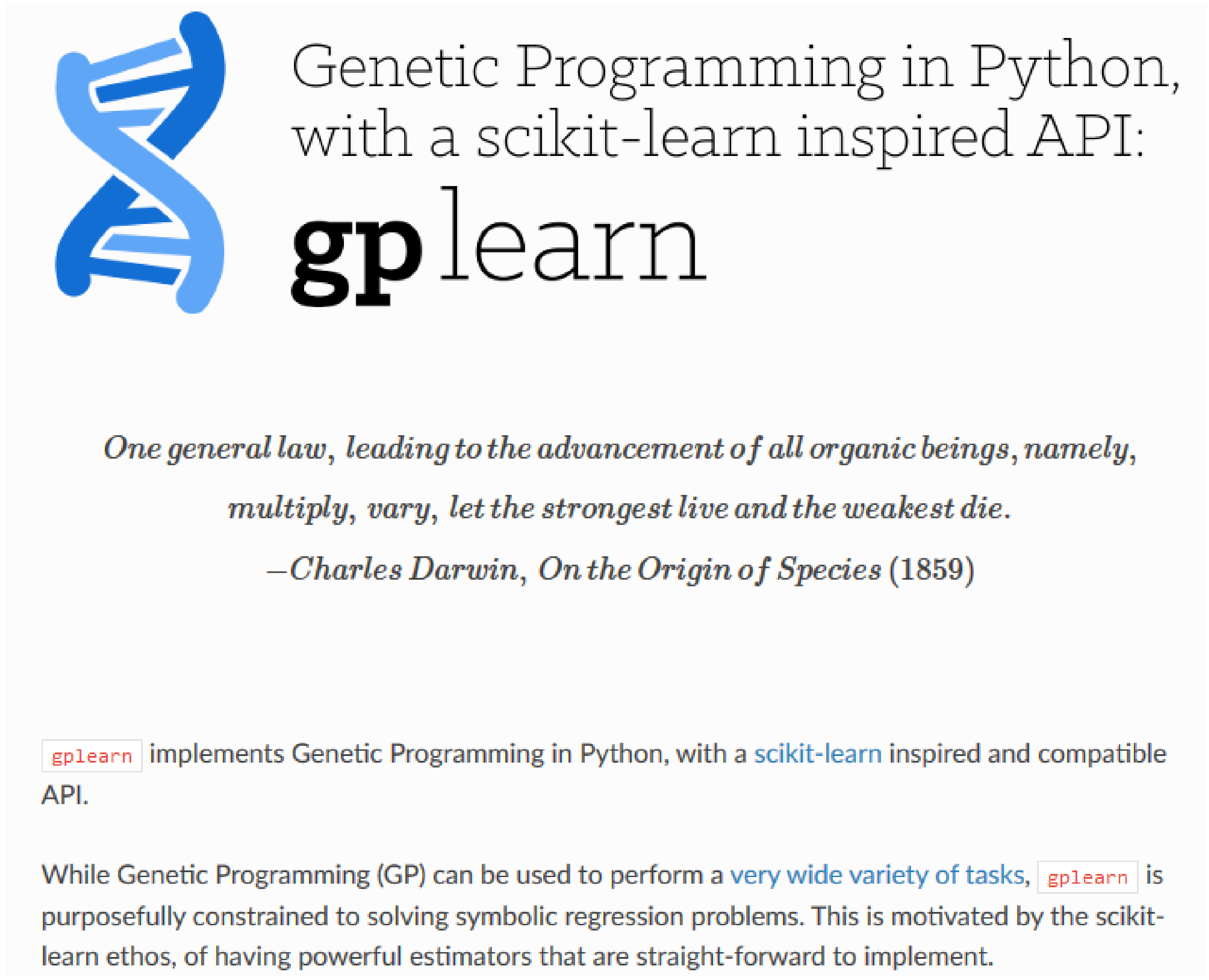
DISTRIBUTED EVOLUTIONARY ALGORITHMS IN PYTHON

https://deap.readthedocs.io/en/master/

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

Genetic Algorithms: …frameworks - gplearn



Genetic Programming in Python, with a scikit-learn inspired API:

gplearn

One general law, leading to the advancement of all organic beings, namely, multiply, vary, let the strongest live and the weakest die.

—*Charles Darwin*, *On the Origin of Species* (1859)

`gplearn` implements Genetic Programming in Python, with a scikit-learn inspired and compatible API.

While Genetic Programming (GP) can be used to perform a very wide variety of tasks, `gplearn` is purposefully constrained to solving symbolic regression problems. This is motivated by the scikit-learn ethos, of having powerful estimators that are straight-forward to implement.

https://gplearn.readthedocs.io/en/stable/

THE SMALL DATA PREDICTIVE MODELING COMPANY

MYDATAMODELS

Genetic Algorithms: BIBLIOGRAPHY

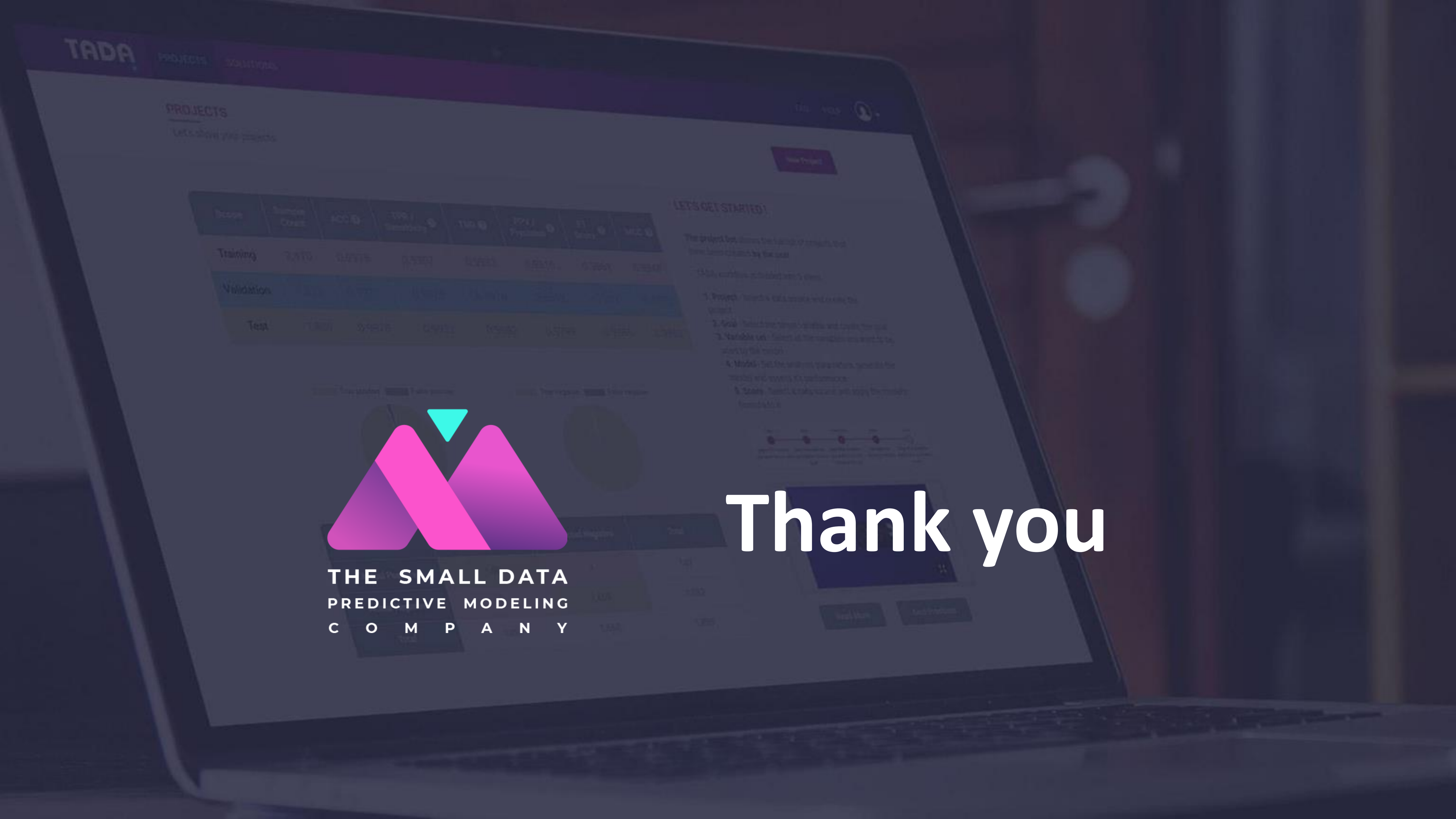
Cited Articles and books

John Holland (Adaptation in Natural and Artificial Systems, 1975)
A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, Second Edition, Springer, ISBN 978-3-662-44873-1, 2015
Stefan Droste, Thomas Jansen, Ingo Wegener, On the analysis of the (1 + 1) evolutionary algorithm , Theoretical Computer Science 276 (2002) 51–81
Santosh Kumar Satpathy, Anirban Mitra, R K Mohanty, Evolutionary Computation: A review on concepts and issues, National Conference On Recent Trends In Soft Computing & its Applications (RTSCA-2K17, 2017)
Hans-Georg Beyera, Hans-Paul Schwefela, Ingo Wegener, How to analyse evolutionary algorithms,
Theoretical Computer Science 287 (2002) 101–130

Used links

Genetic Algorithms, E D Goodman, Michigan State Univ: https://slideplayer.com/slide/5339767/
https://blog.overops.com/how-to-solve-tough-problems-using-genetic-algorithms/
https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_introduction.htm
https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3
https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_parent_selection.htm
https://deap.readthedocs.io/en/master/
https://cran.r-project.org/web/views/Optimization.html
http://geneticprogramming.com/
https://gplearn.readthedocs.io/en/stable/

Links
general resources



THE SMALL DATA
PREDICTIVE MODELING
COMPANY

Thank you