Intro

Competitions' concepts

Data

Model

Submission

Evaluation

Leaderboard

Competitions' concepts

Data

Model

Submission

Evaluation

Leaderboard

Example: data

Overview | Data | Kernels | Discussion | Leaderboard | Rules | Submit Predictions

Training Data

- properties_2016.csv....
- sample_submission.cs...
- train_2016_v2.csv.zi...
- zillow_data_dictiona...

zillow_data_dictionary.xlsx.zip 15.74 KB

Download

Data Introduction

In this competition, Zillow is asking you to predict the log-error between their Zestimate and the actual sale price, given all the features of a home. The log error is defined as

$$logerror = log(Zestimate) - log(SalePrice)$$

and it is recorded in the transactions file **train.csv**. In this competition, you are going to predict the logerror for the months in Fall 2017. Since all the real estate transactions in the U.S. are publicly available, we will close the competition (no longer accepting submissions) before the evaluation period begins.

Train/Test split

- You are provided with a full list of real estate properties in three counties (Los Angeles, Orange and Ventura, California) data in 2016.
- The train data has all the transactions before October 15, 2016, plus some of the transactions after October 15, 2016.

Competitions' concepts

Data

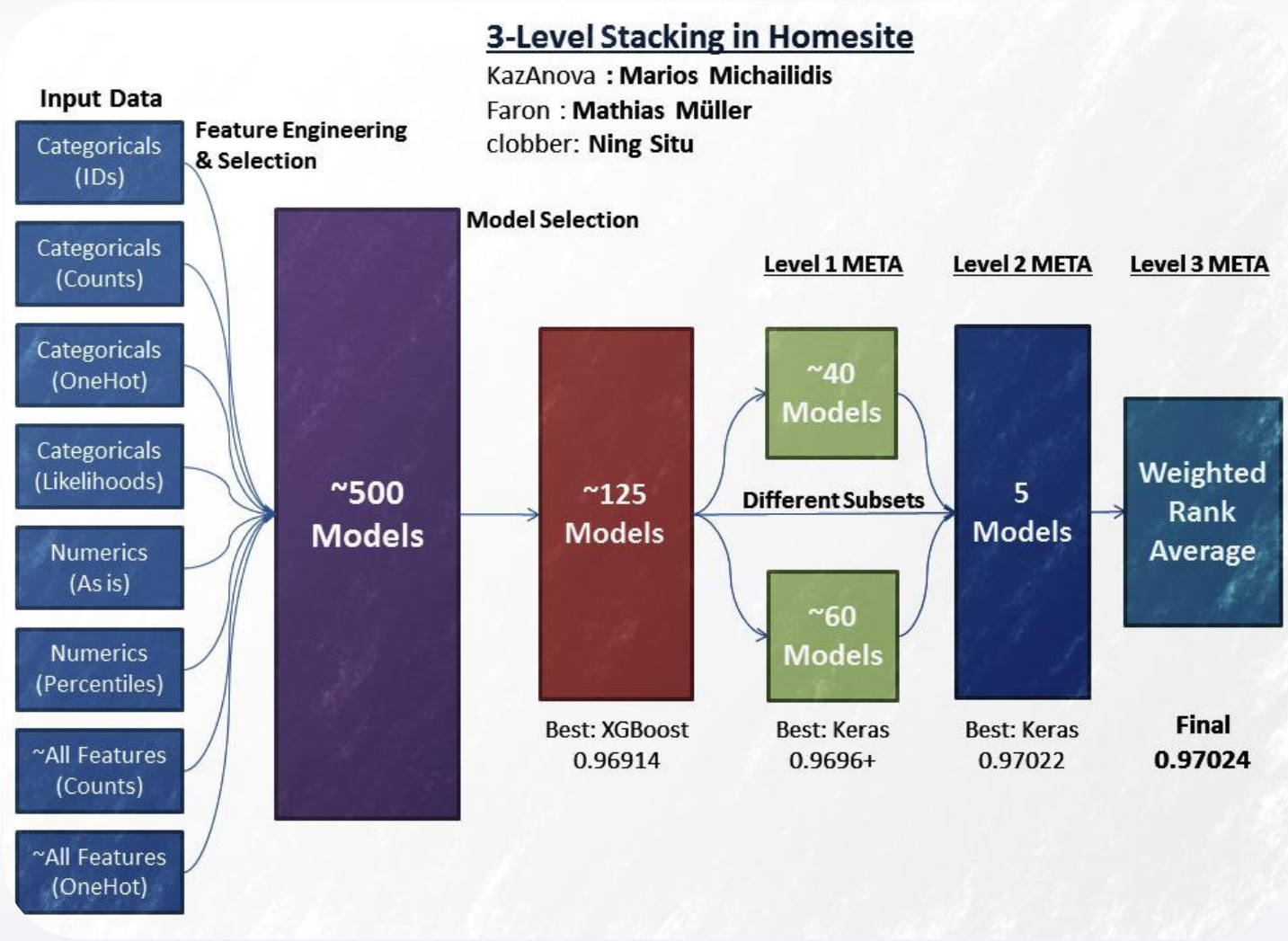
Model

Submission

Evaluation

Leaderboard

Model Example



http://blog.kaggle.com/2016/04/08/homesite-quote-conversion-winners-write-up-1st-place-kazanova-faron-clobber

Competitions' concepts

Data

Model

Submission

Evaluation

Leaderboard

Submission

Usually you are asked to submit only predictions.

Submission

Usually you are asked to submit only predictions.

Sample submission usually looks like:

```
ParcelId,201610,201611,201612,201710,201711,201712
10754147,0.1234,1.2234,-1.3012,1.4012,0.8642-3.1412
10759547,0,0,0,0,0,0
etc.
```

Screenshot kaggle.com

Competitions' concepts

Data

Model

Submission

Evaluation

Leaderboard

Evaluation function

You need to know how good is your model.

The quality of the model is defined by evaluation function:

(predictions, right answers) -> score

Evaluation function

Examples:

- Accuracy
- Logistic loss
- AUC
- RMSE
- MAE

Competitions' concepts

Data

Model

Submission

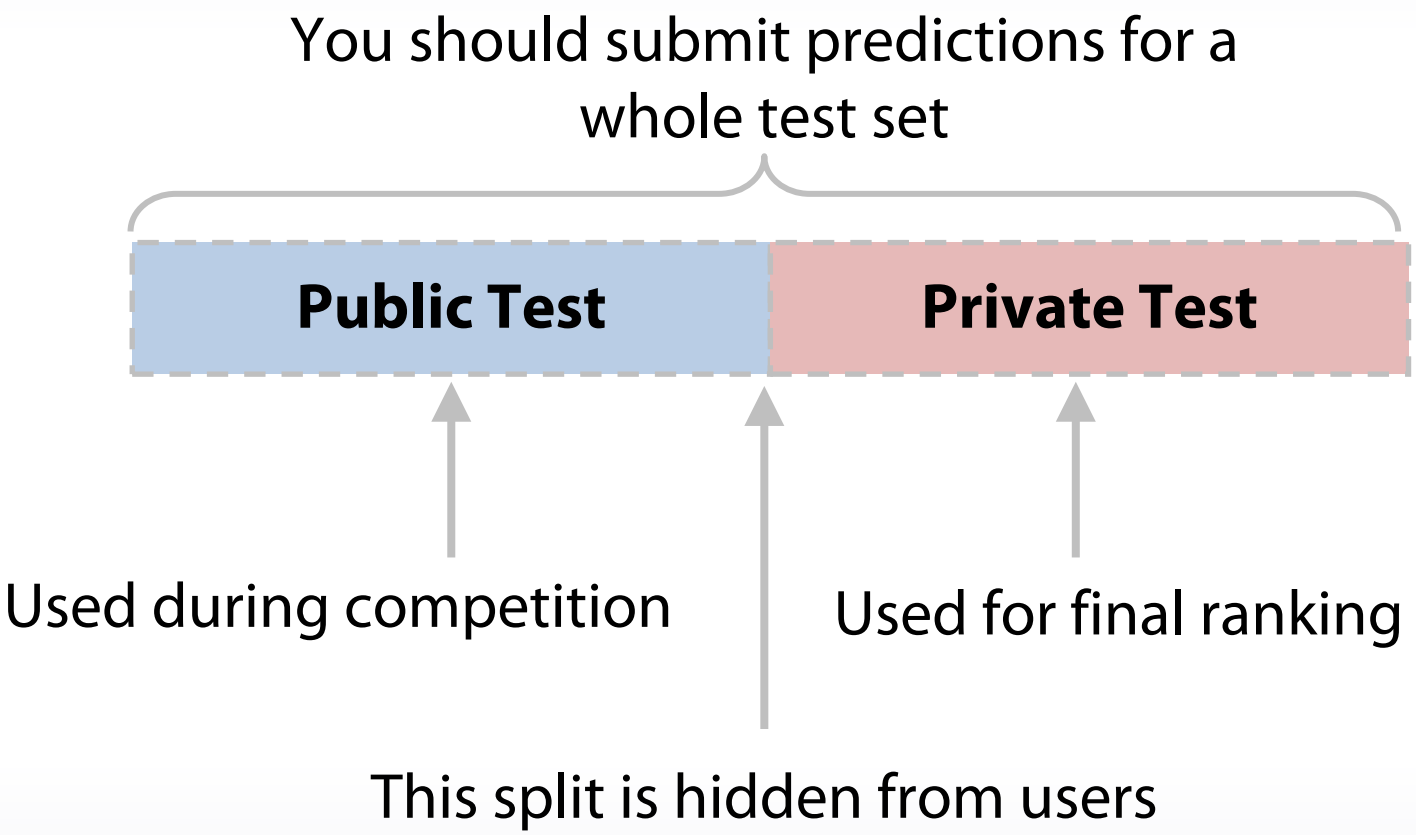
Evaluation

Leaderboard

Leaderboard

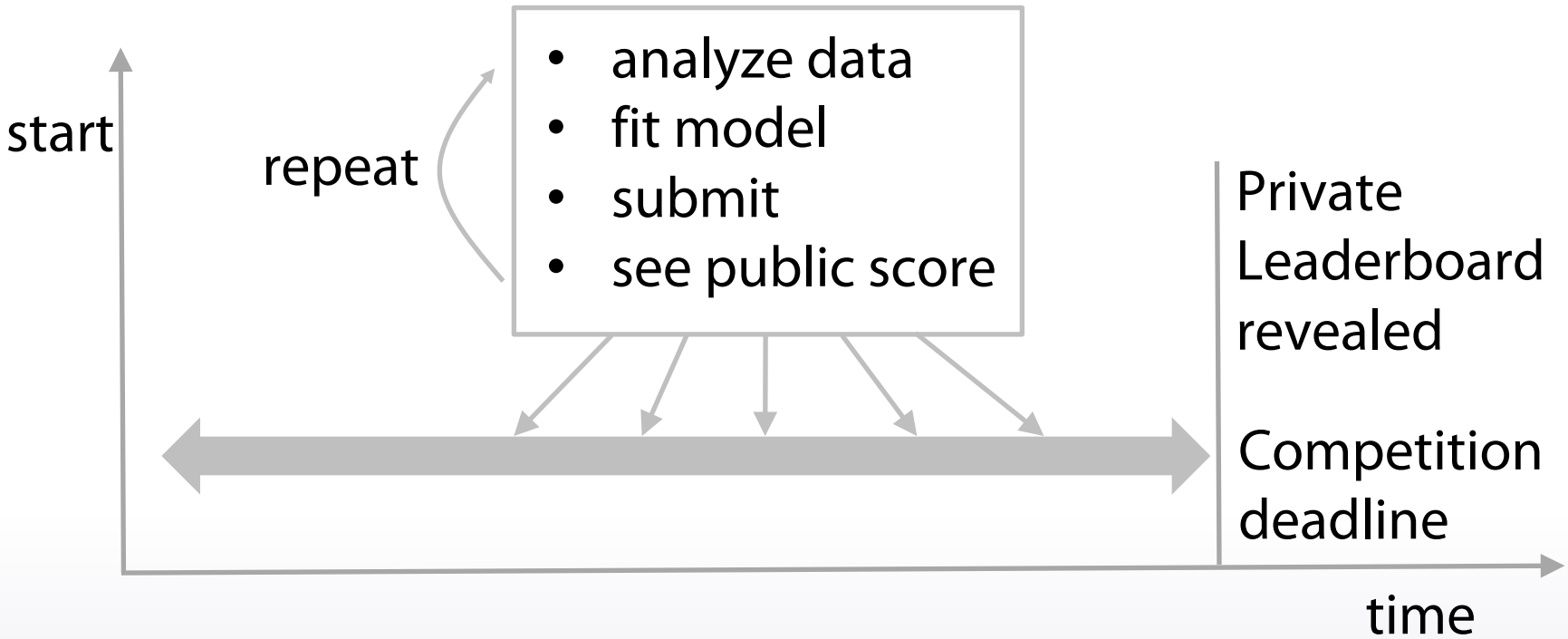
#	△pub	Team Name	Score	Entries	Last
1	▲2	Chenglong Chen	0.72189	160	2y
2	▲4	Mikhail & Stanislav & Dmitry	0.71871	83	2y
3	▼2	Quartet	0.71861	279	2y
4	▲1	Shize & Shail & Phil	0.71802	252	2y
5	▲8	I love Phở Bò	0.71700	48	2y
6	▼2	Gzs_iceberg	0.71681	122	2y
7	▲1	YDM	0.71374	283	2y
8	▲10	A & A & G	0.71297	229	2y
9	▲7	ë	0.71265	96	2y
10	▲4	Alexander D'yakonov (PZAD, ...	0.71262	93	2y
11	▼9	SearchSearchSearch	0.71022	58	2y
12	▲8	woshialex	0.70889	52	2y
13	▲43	Alexander Ryzhkov (PZAD, Ru...	0.70777	64	2y
14	▼7	Jianmin Sun	0.70711	145	2y
15	▼6	I survived Glastonbury (just)	0.70705	119	2y

Public/Private Tests



Example of competition mechanics

Only public leaderboard is available at this stage



Competitions' concepts

Data

Model

Submission

Evaluation

Leaderboard

Platforms:

- Kaggle
- DrivenData
- CrowdAnalityx
- CodaLab
- DataScienceChallenge.net
- Datascience.net
- Single-competition sites (like KDD, VizDooM)

Why to participate?

- Great opportunity for learning and networking

Why to participate?

- Great opportunity for learning and networking
- Interesting non-trivial tasks and state-of-the-art approaches

Why to participate?

- Great opportunity for learning and networking
- Interesting non-trivial tasks and state-of-the-art approaches
- A way to get famous inside data science community

Why to participate?

- Great opportunity for learning and networking
- Interesting non-trivial tasks and state-of-the-art approaches
- A way to get famous inside data science community
- A way to earn some money

Conclusion

- Main concepts:
  - Data
  - Model
  - Submission
  - Evaluation
  - Leaderboard
- Competition platforms
- Reasons for participating

Recap

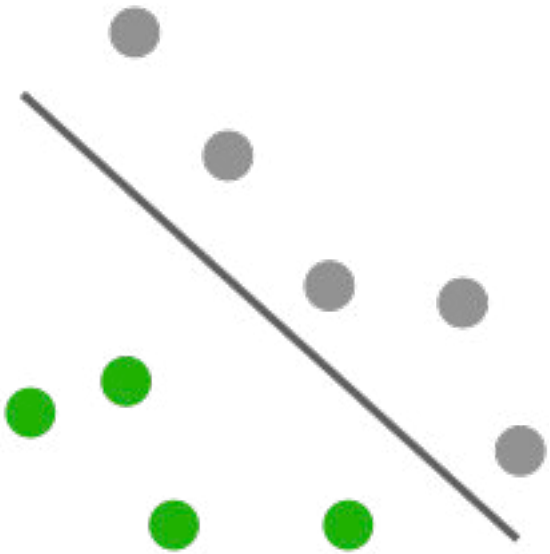
Families of ML algorithms

- Linear
- Tree-based
- kNN
- Neural Networks

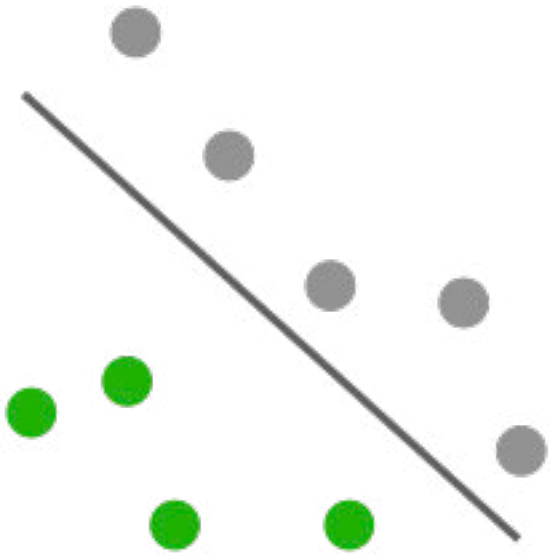
Linear model

Linear model

Linear model



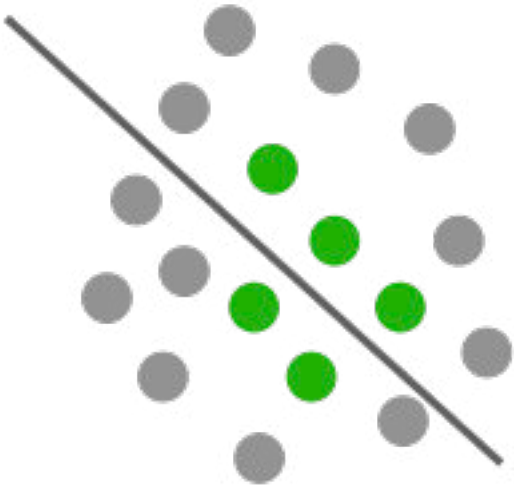
Linear model



Examples:

- Logistic Regression
- Support Vector Machines

Linear model



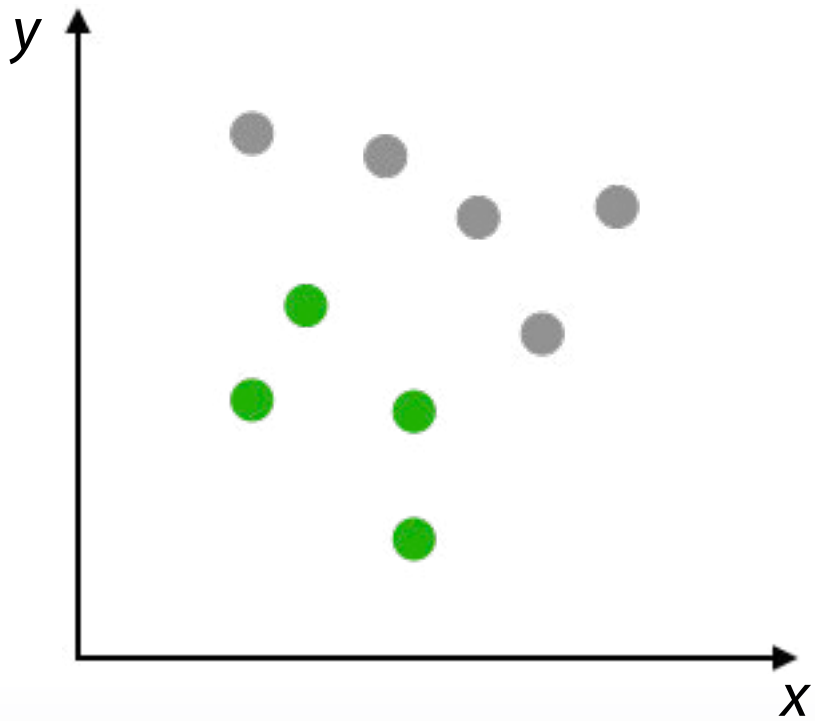
Linear model

scikit learn

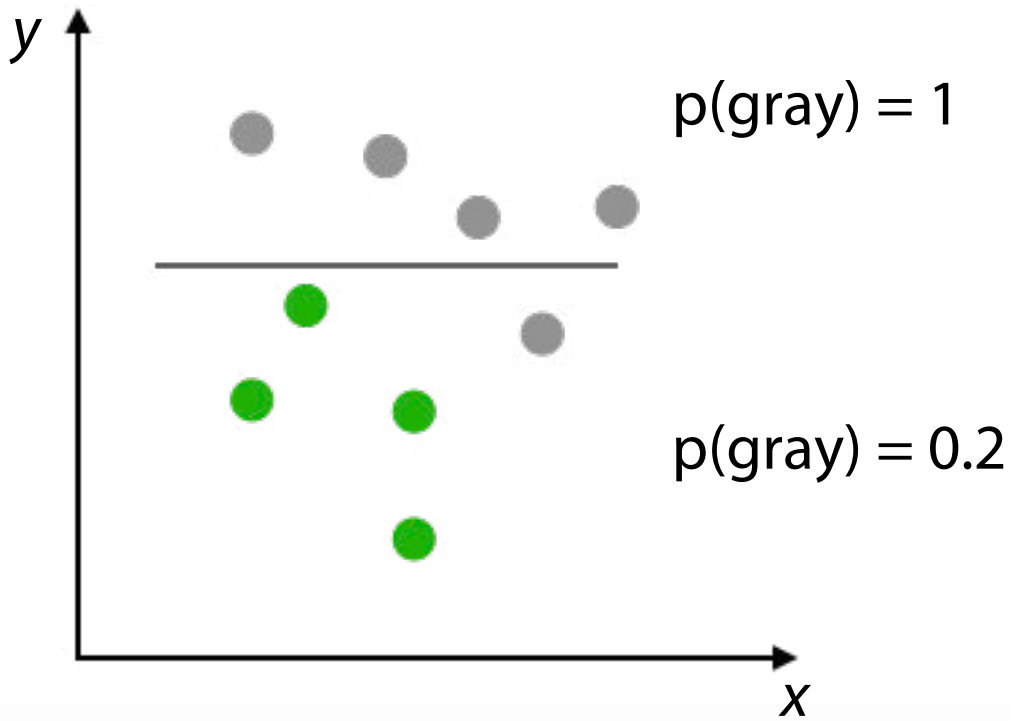
VOWPAL WABBIT

Tree-based: Decision Tree, Random Forest, GBDT

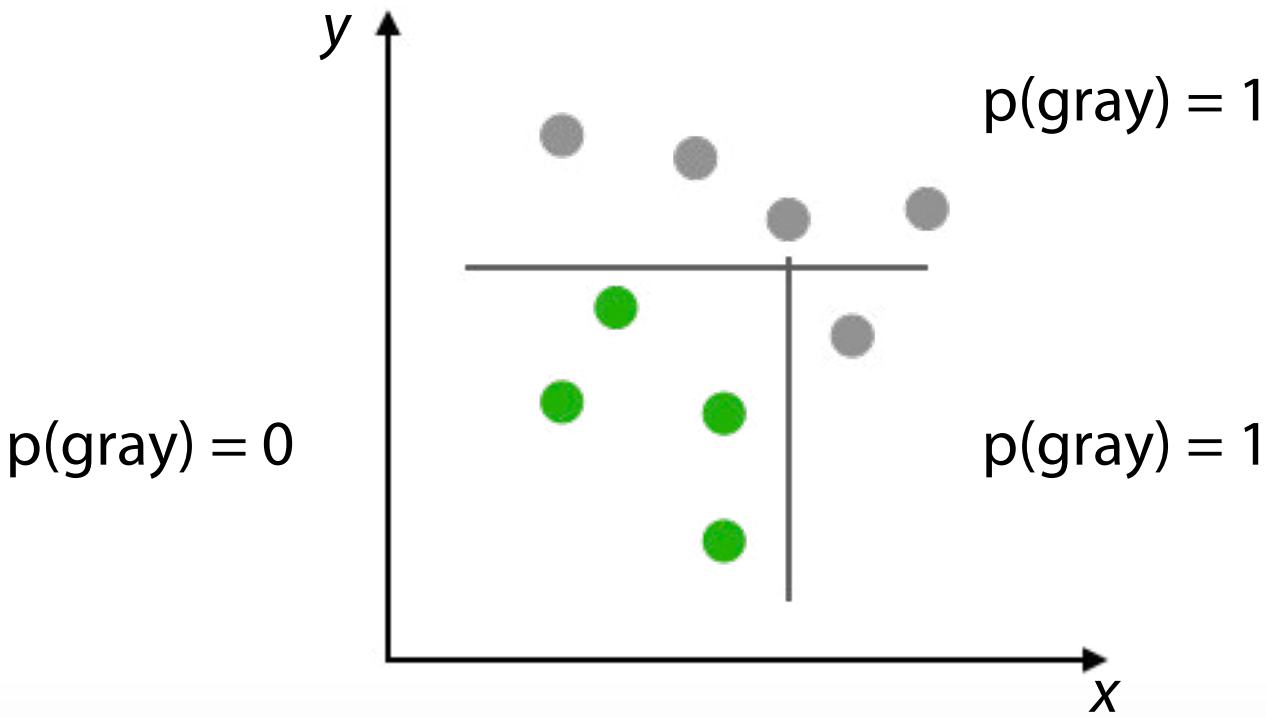
Tree-based: Decision Tree, Random Forest, GBDT



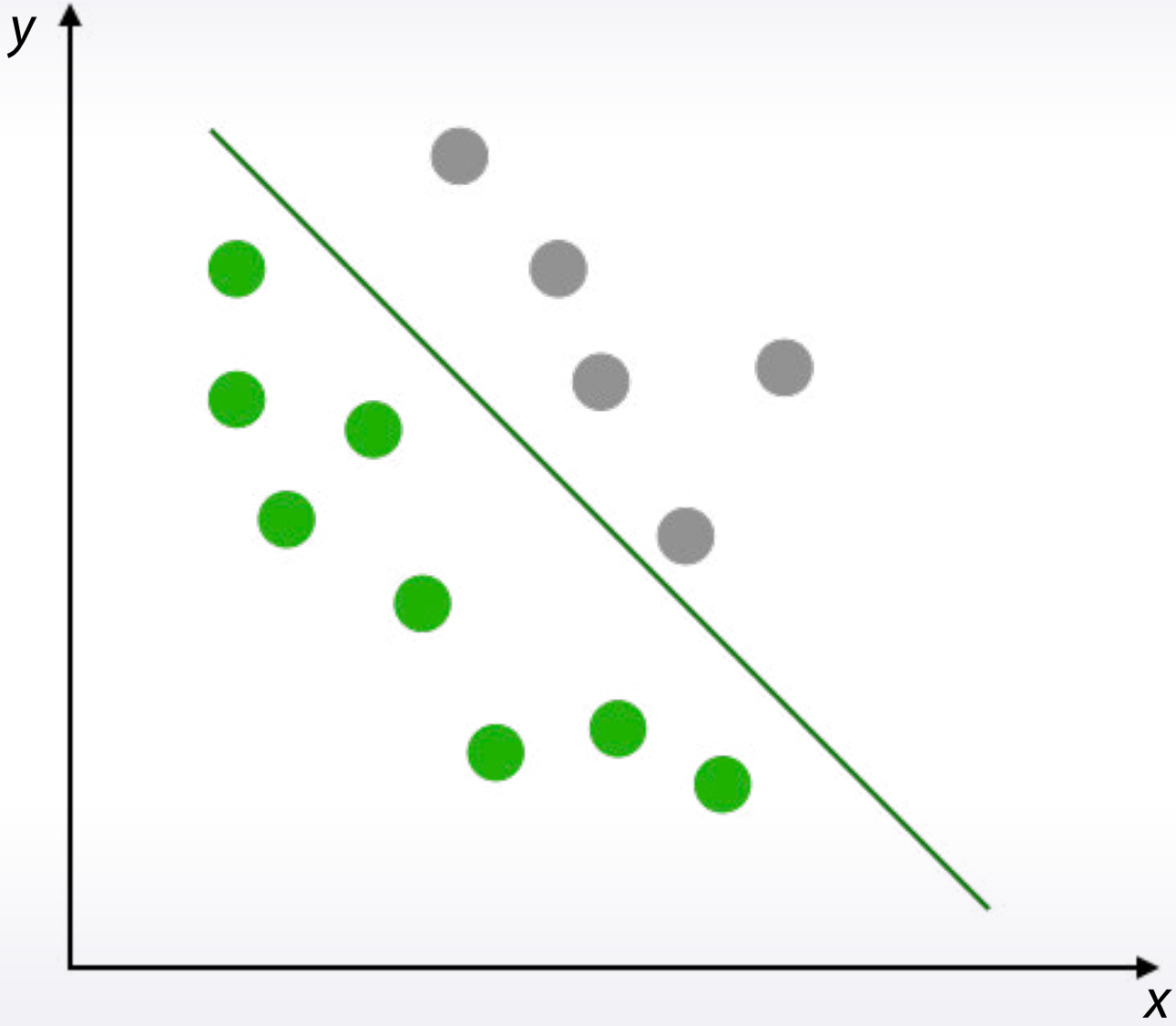
Tree-based: Decision Tree, Random Forest, GBDT



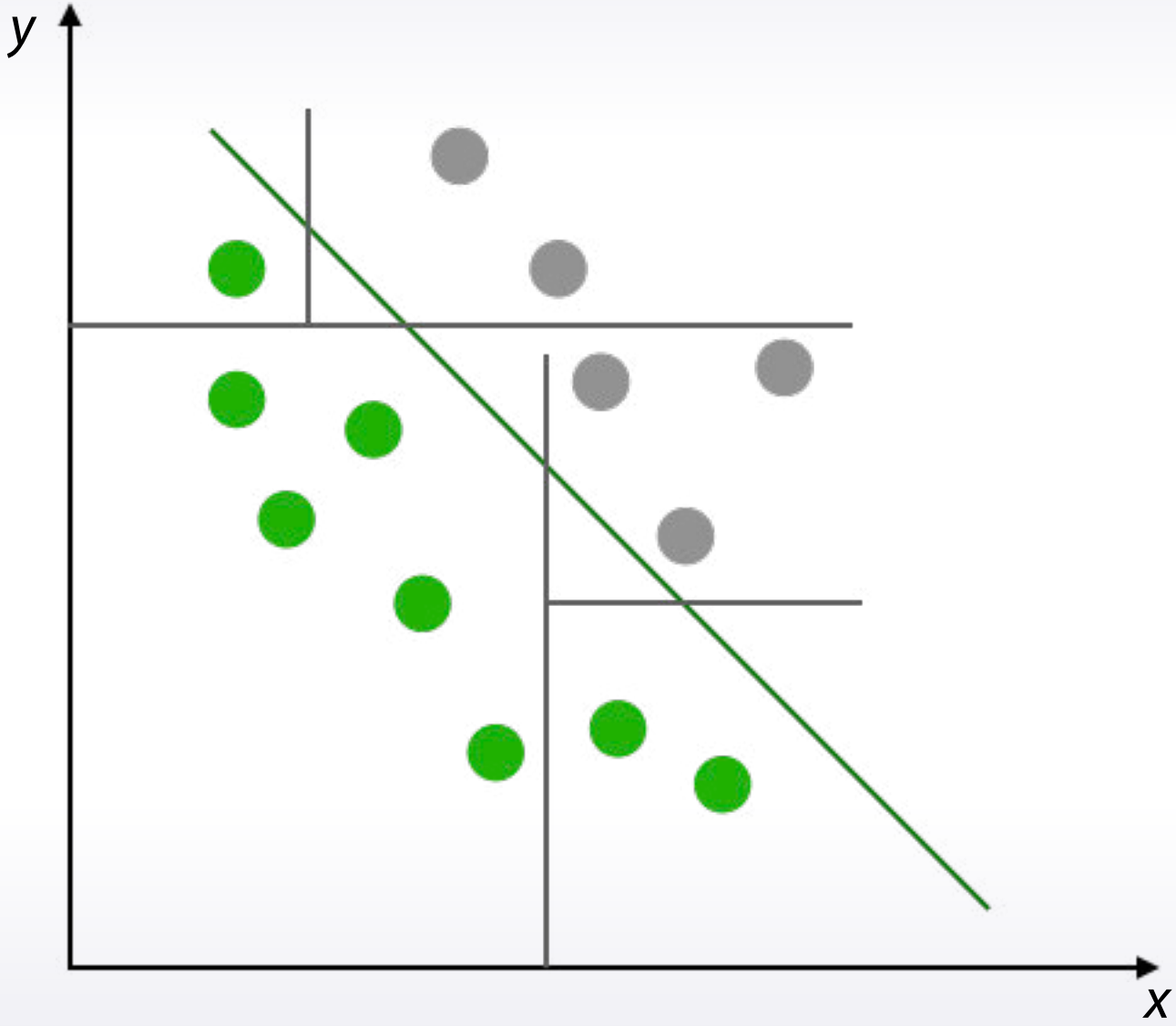
Tree-based: Decision Tree, Random Forest, GBDT



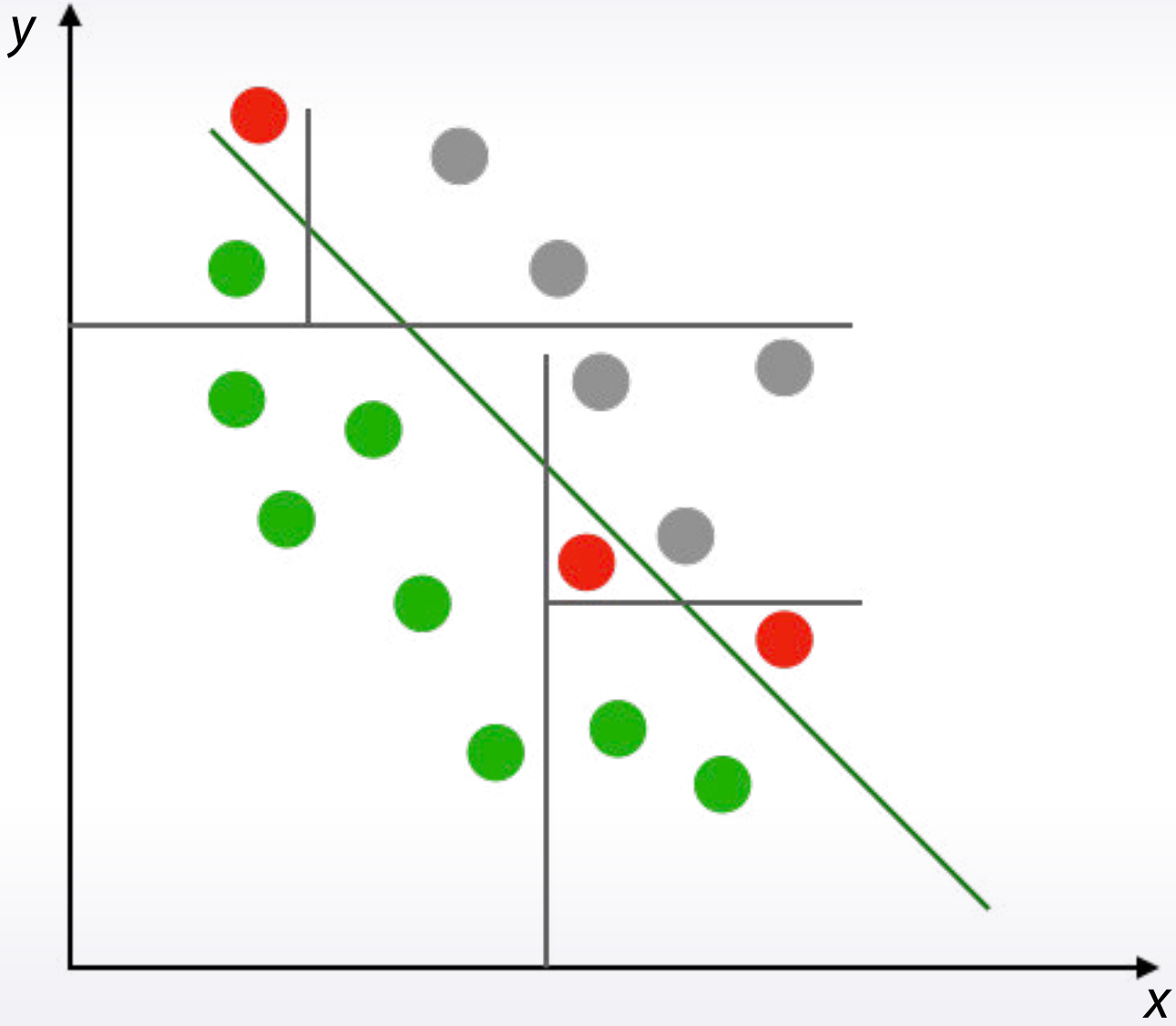
Tree-based: Decision Tree, Random Forest, GBDT



Tree-based: Decision Tree, Random Forest, GBDT



Tree-based: Decision Tree, Random Forest, GBDT



Tree-based methods

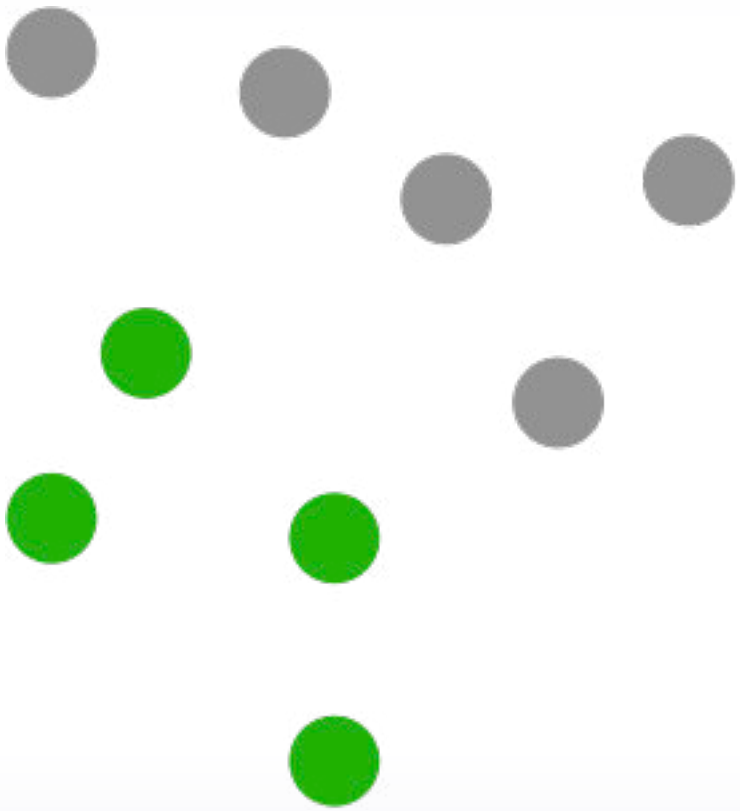
scikit learn

dmlc XGBoost

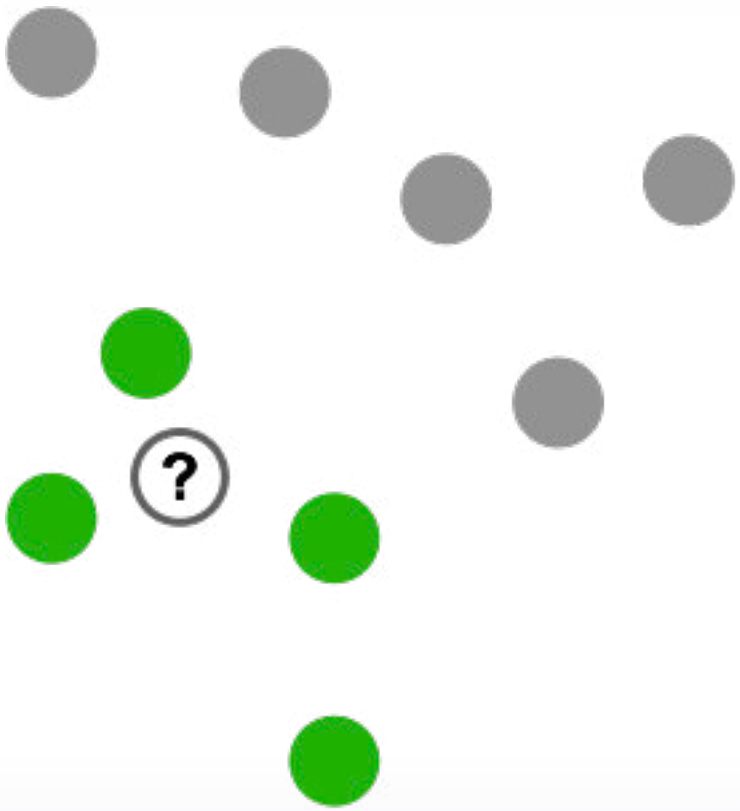
Microsoft / LightGBM

kNN-based methods

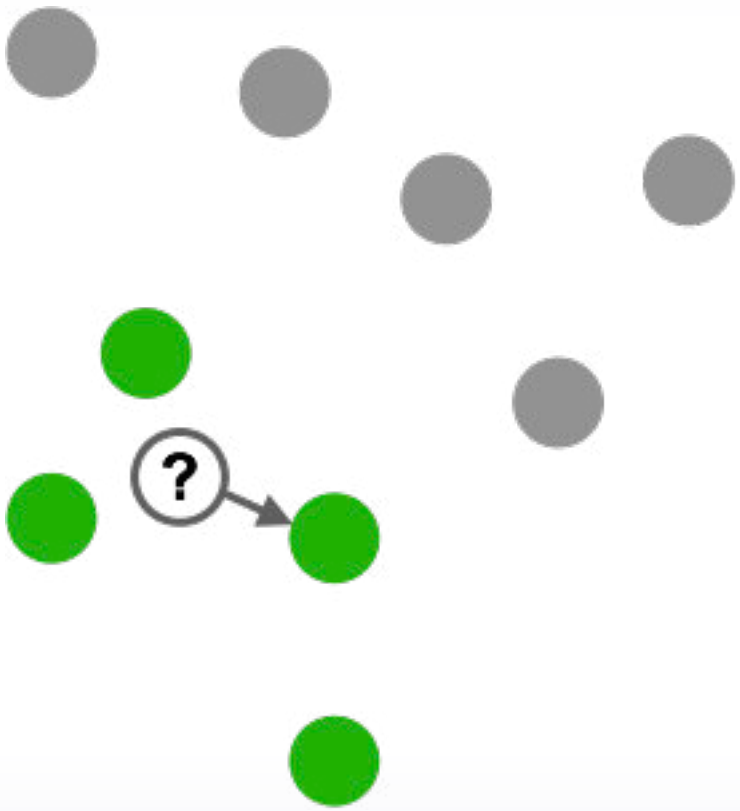
kNN-based methods



kNN-based methods



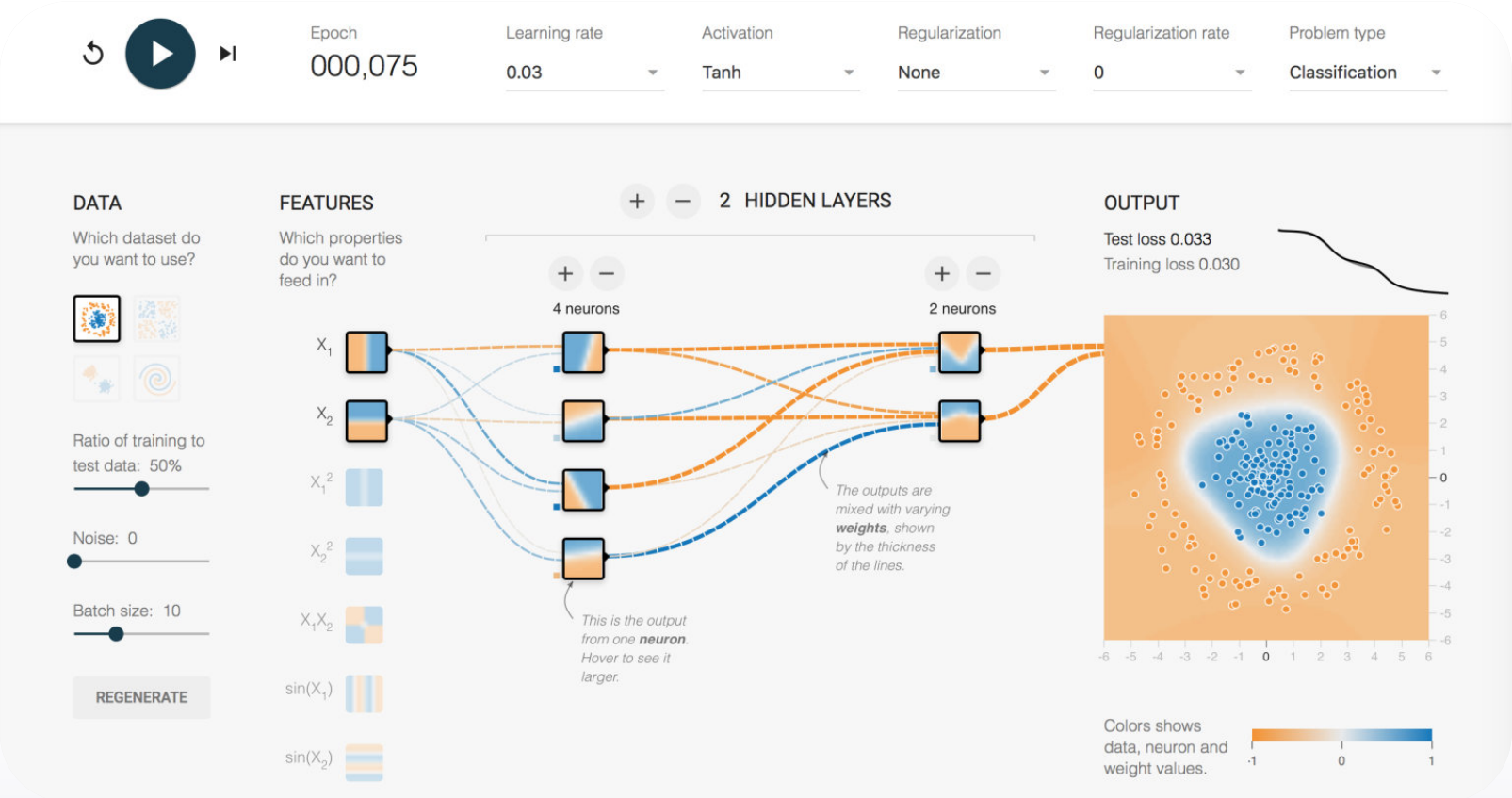
kNN-based methods



kNN-based methods

scikit learn

Neural Networks



Tensorflow Playground, http://playground.tensorflow.org

Neural Networks

TensorFlow

K

dmlc mxnet

PYTORCH

Lasagne

No Free Lunch Theorem

No Free Lunch Theorem

"Here is no method which **outperforms all others for all tasks**"

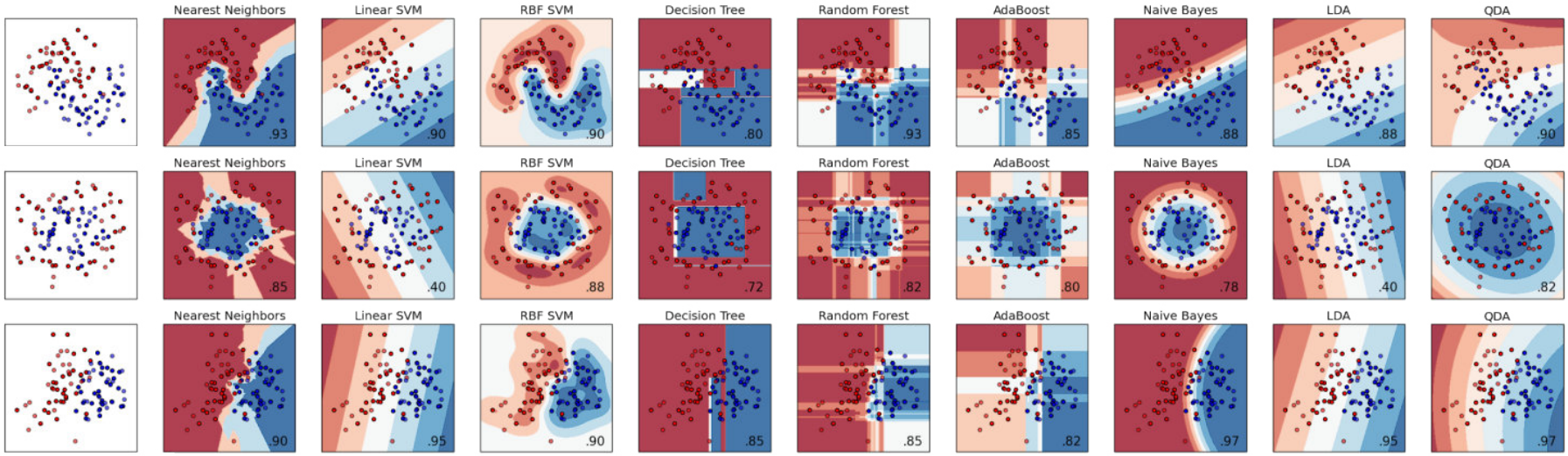
No Free Lunch Theorem

"Here is no method which **outperforms all others for all tasks**"

or

"For every method **we can construct a task** for which **this particular method will not be the best**"

Decision surfaces



Classifier comparison, http://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html

Conclusion

- There is no “silver bullet” algorithm

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes
- k-NN methods heavy rely on how to measure points “closeness”

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes
- k-NN methods heavy rely on how to measure points “closeness”
- Feed-forward NNs produce smooth non-linear decision boundary

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes
- k-NN methods heavy rely on how to measure points “closeness”
- Feed-forward NNs produce smooth non-linear decision boundary

The most powerful methods are
Gradient Boosted Decision Trees and **Neural Networks**.
But you shouldn’t underestimate the others

Hardware/Software setup

Hardware

- Most of competitions (expect image-based) can be solved on:
  - High-level laptop
  - 16+ gb ram
  - 4+ cores
- Quite good setup:
  - Tower PC
  - 32+ gb ram
  - 6+ cores

Hardware

Really important things:

- **RAM**

  If you can keep data in memory — everything will be much easier
- **Cores**

  More cores you have — more (or faster) experiments you can do
- **Storage**

  SSD is crucial if you work with images or big datasets with a lot of small pieces

Cloud resources

Cloud platforms can provide you with a computational resources.

There are several cloud options:

- Amazon AWS
- Microsoft Azure
- Google Cloud

Cloud resources

Cloud platforms can provide you with a computational resources.

There are several cloud options:

- Amazon AWS spot option!
- Microsoft Azure
- Google Cloud

Software: language

Most of competitors use Python data science software stack.

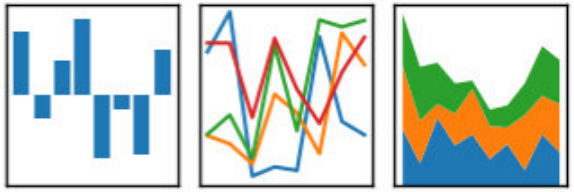
Basic stack

Most of competitors use Python data science software stack.

NumPy

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



scikit learn

matplotlib

IDE

IP[y]:
IPython

jupyter

Special packages

dmlc XGBoost

Microsoft / LightGBM

Keras

danielfrg / tsne
forked from osdf/py_bh_tsne

External tools

VOWPAL WABBIT

srendle / libfm

guestwalk / libffm

baidu / fast_rgf

Conclusion

- Anaconda works out-of-box
- Proposed setup is not the only one, but most common
- Don’t overestimate role of hardware\software