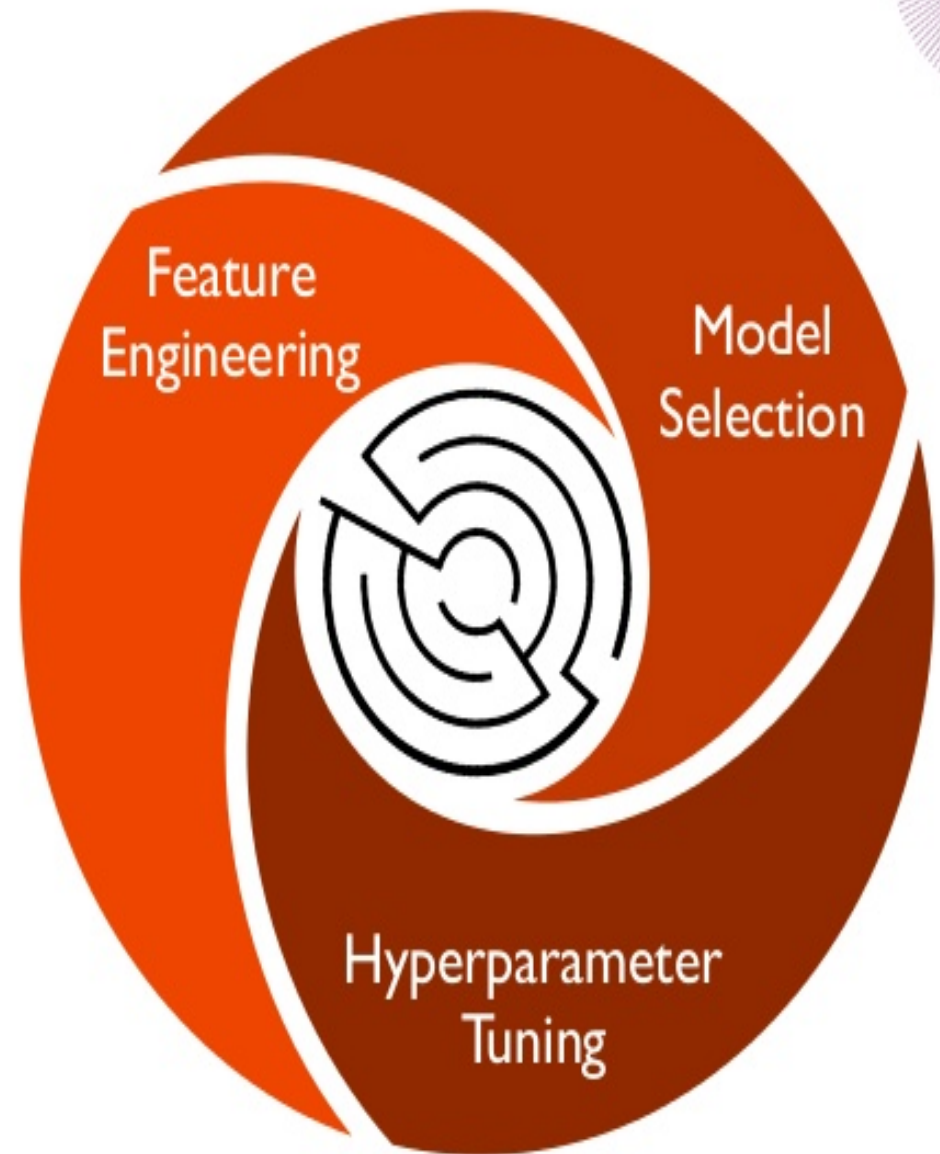# ARIMO
Data Intelligence for All

Automating Data Science on Spark:
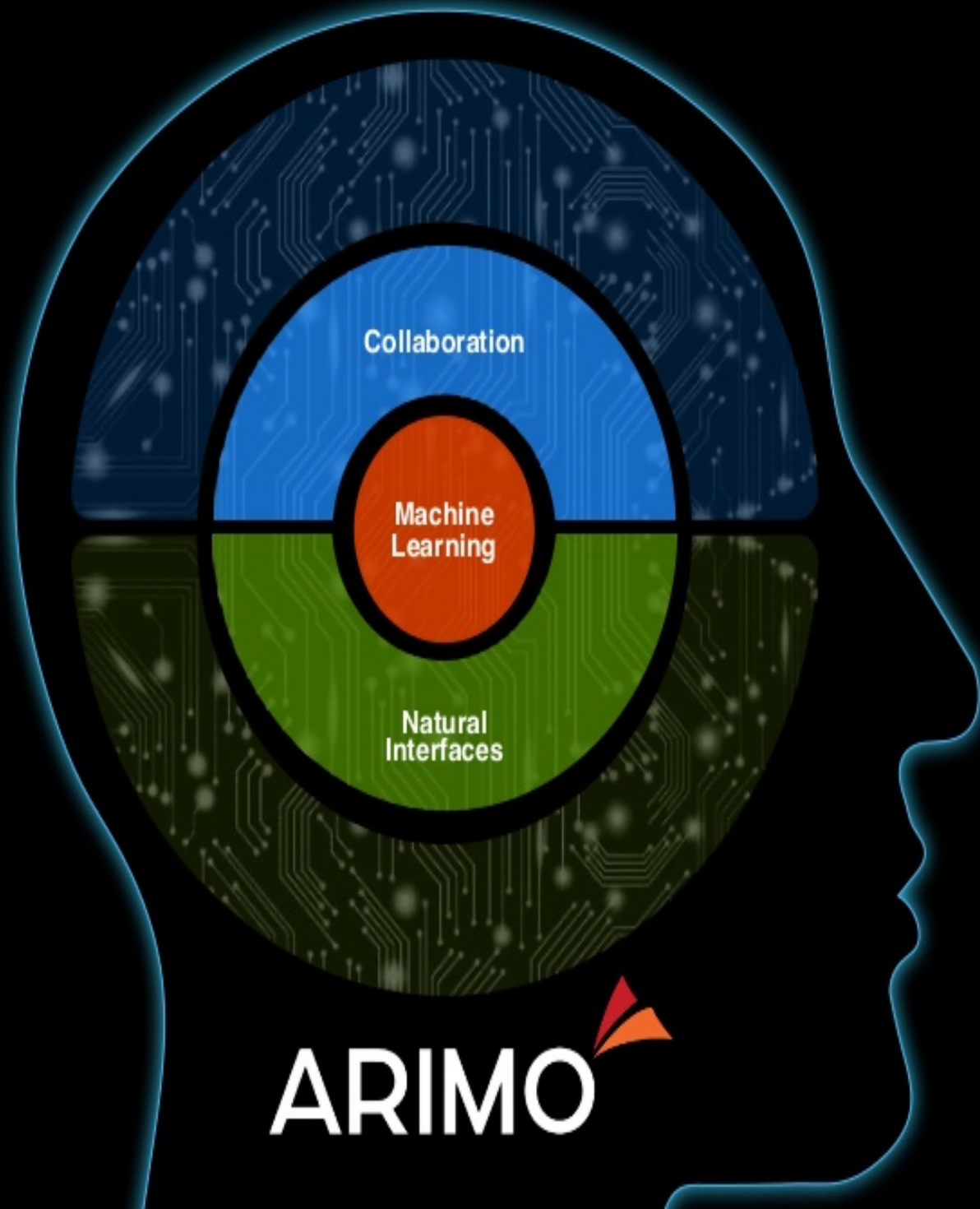## A Bayesian Approach

Vu Pham, Huan Dao, Christopher Nguyen

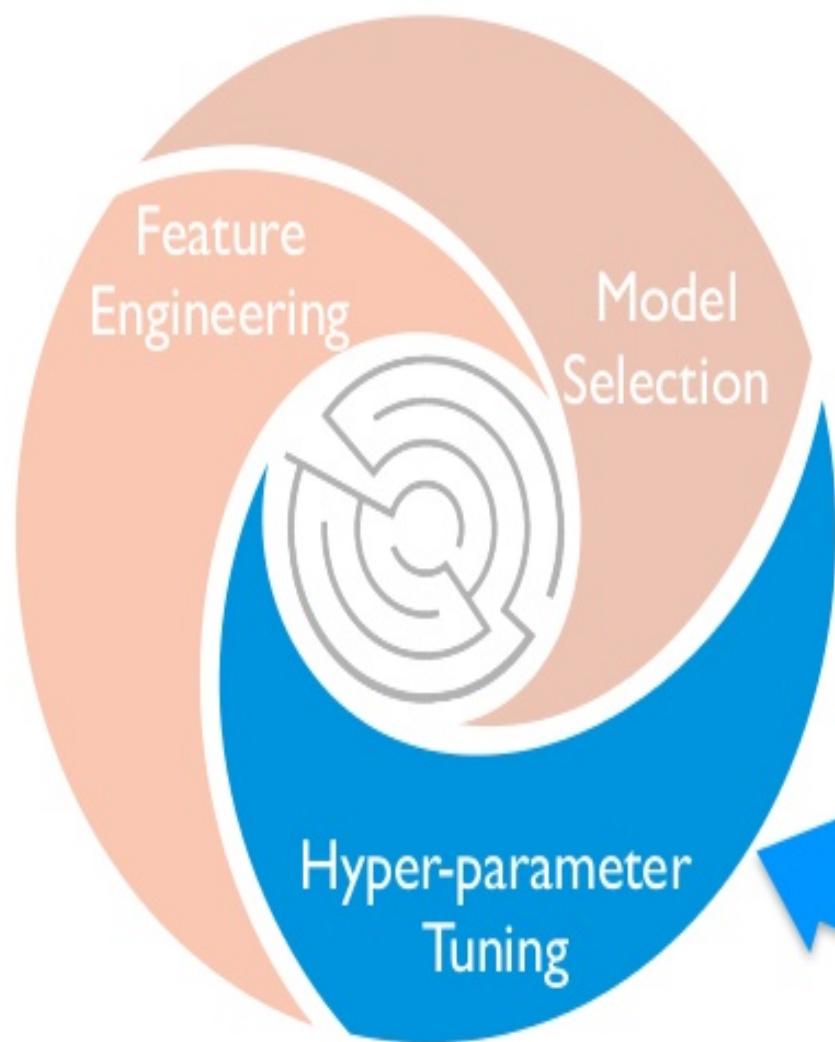San Francisco, June 8th 2016

Spark summit

SPARK SUMMIT 2016

Agenda

1. For Hyper-parameter Tuning

2. For "Automating" Data Science on Spark

3. Experiments

Bayesian Optimization
for Hyper-parameter Tuning

What if...

Feature Engineering

Model Selection

Hyper-parameter Tuning

We can *automate this?*

Hyper-parameters

K-means	K
Neural Network	# of layers, dropout, momentum…
Random Forest	Feature set, # of trees, max depth…
SVM	Regularization term (C)
Gradient Descent	Learning rate, number of iterations…

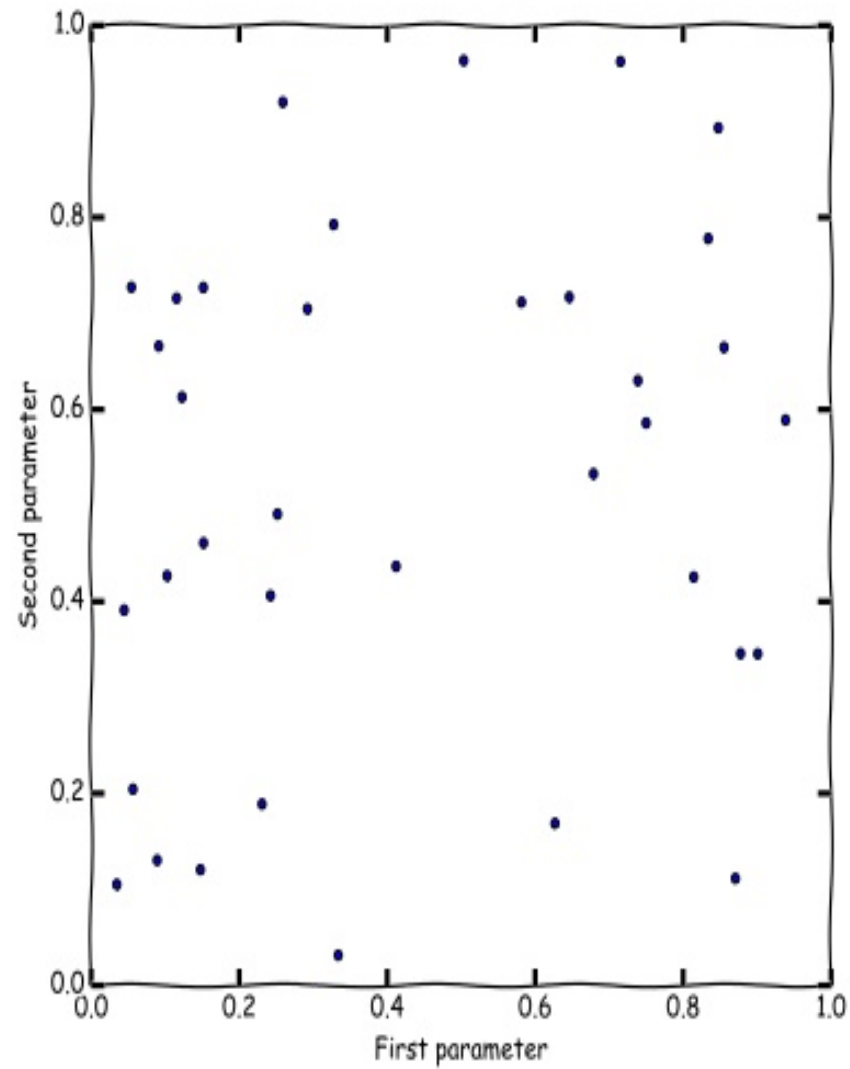# Manual Search
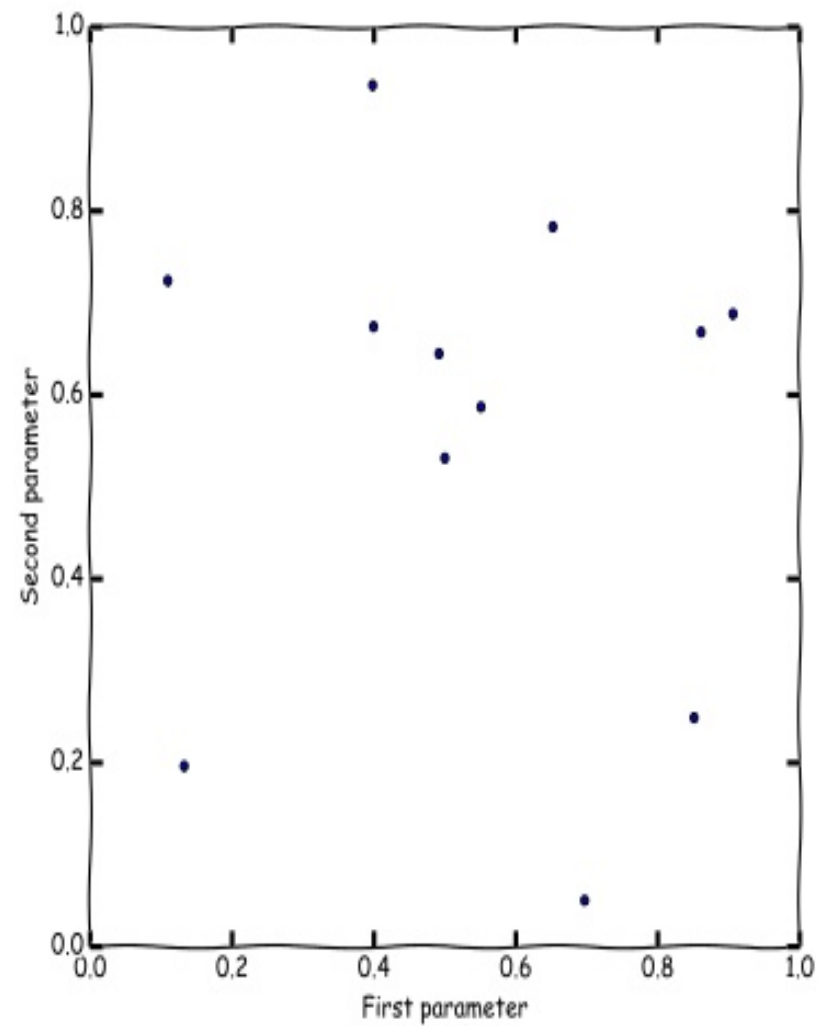
Grid Search
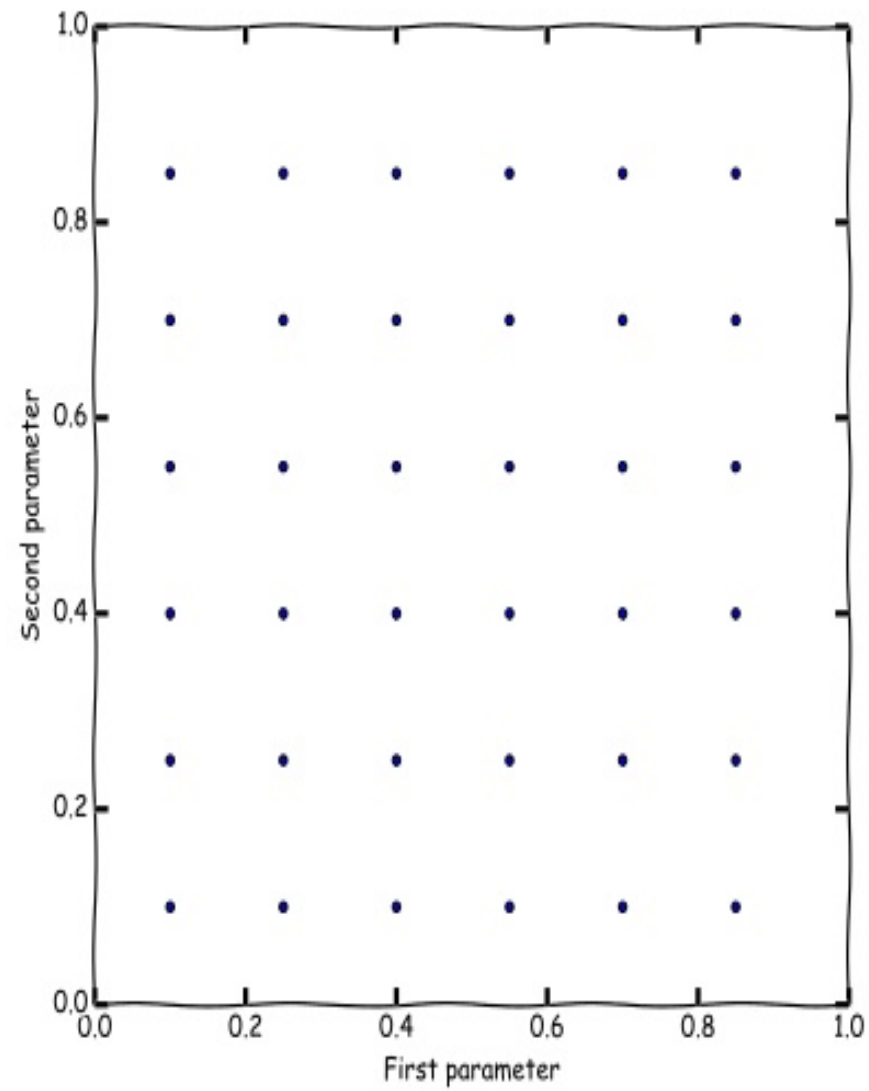
Random Search

Hyper-parameters tuning

Maximize a utility functions over hyper-parameter space:

$$U: \quad S \longrightarrow \mathbb{R}$$

$$c \longmapsto U(c)$$

Hyper-parameter Space

Performance on Validation Set
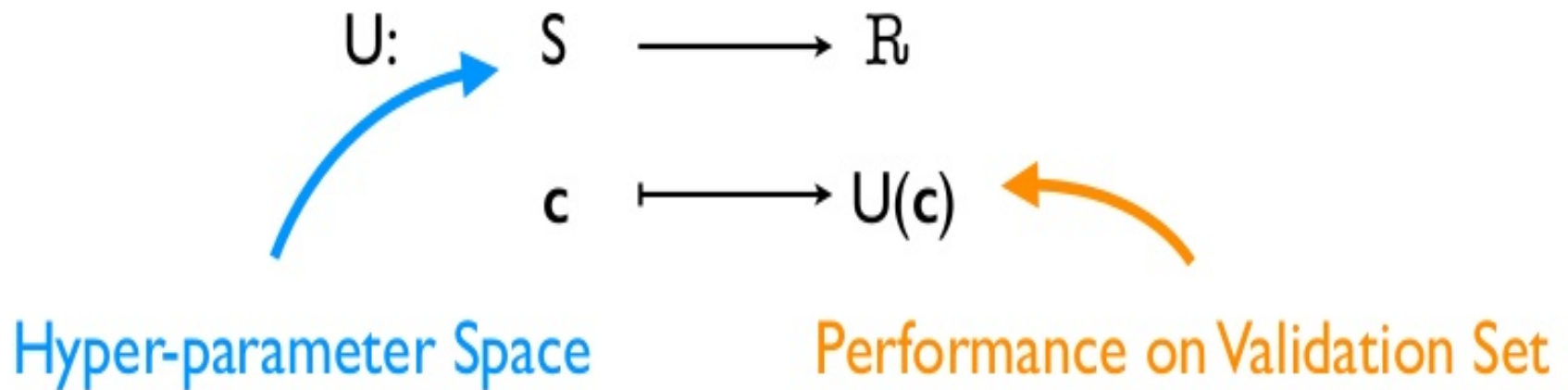
How to intelligently select the next configuration?

(Given the observations in the past)

Bayesian Inference
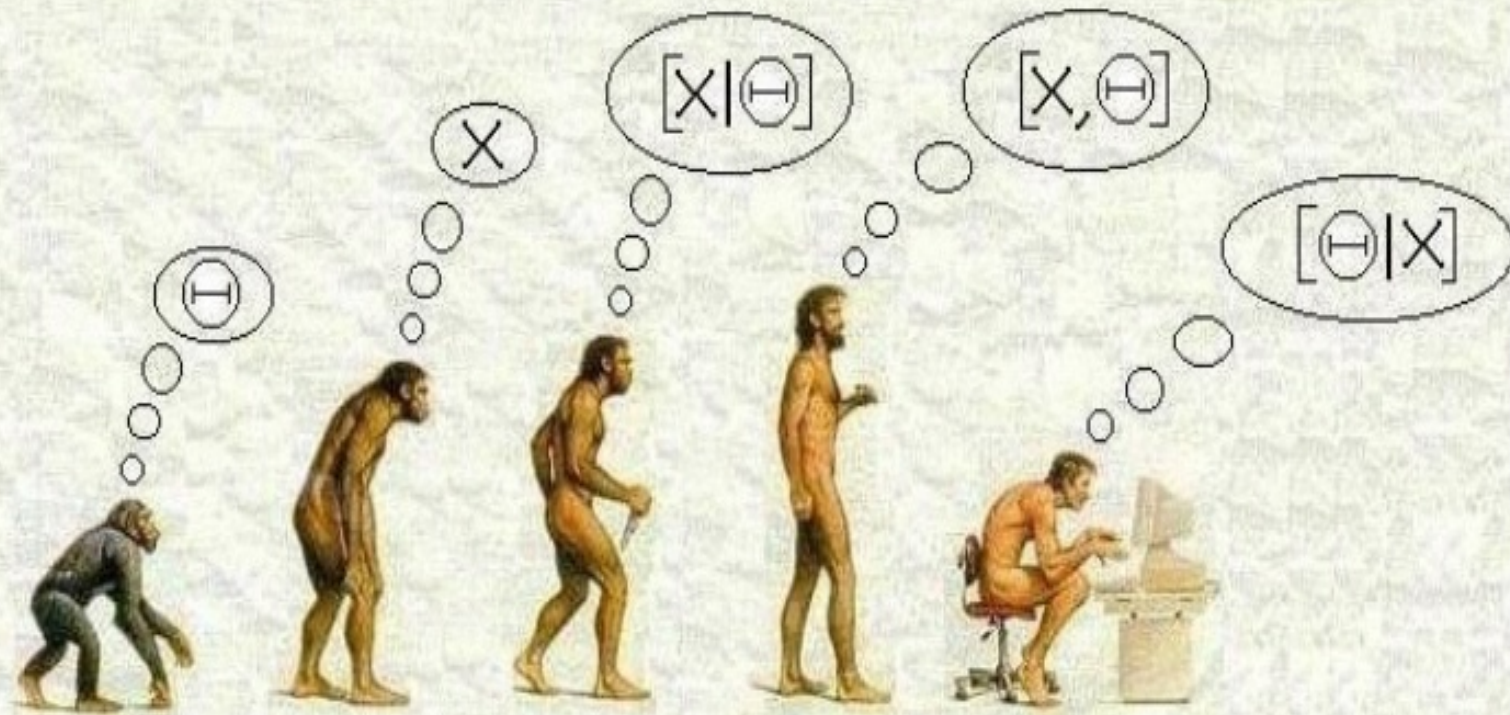
Bayesian Optimization Explained

1. Incorporate a **prior** over the space of possible objective functions (GP)

2. Combine the **prior** with the **likelihood** to obtain a **posterior** over function values given observations

3. Select next configuration to evaluate based on the **posterior**

   - According to an *acquisition function*

Bayesian Optimization Explained
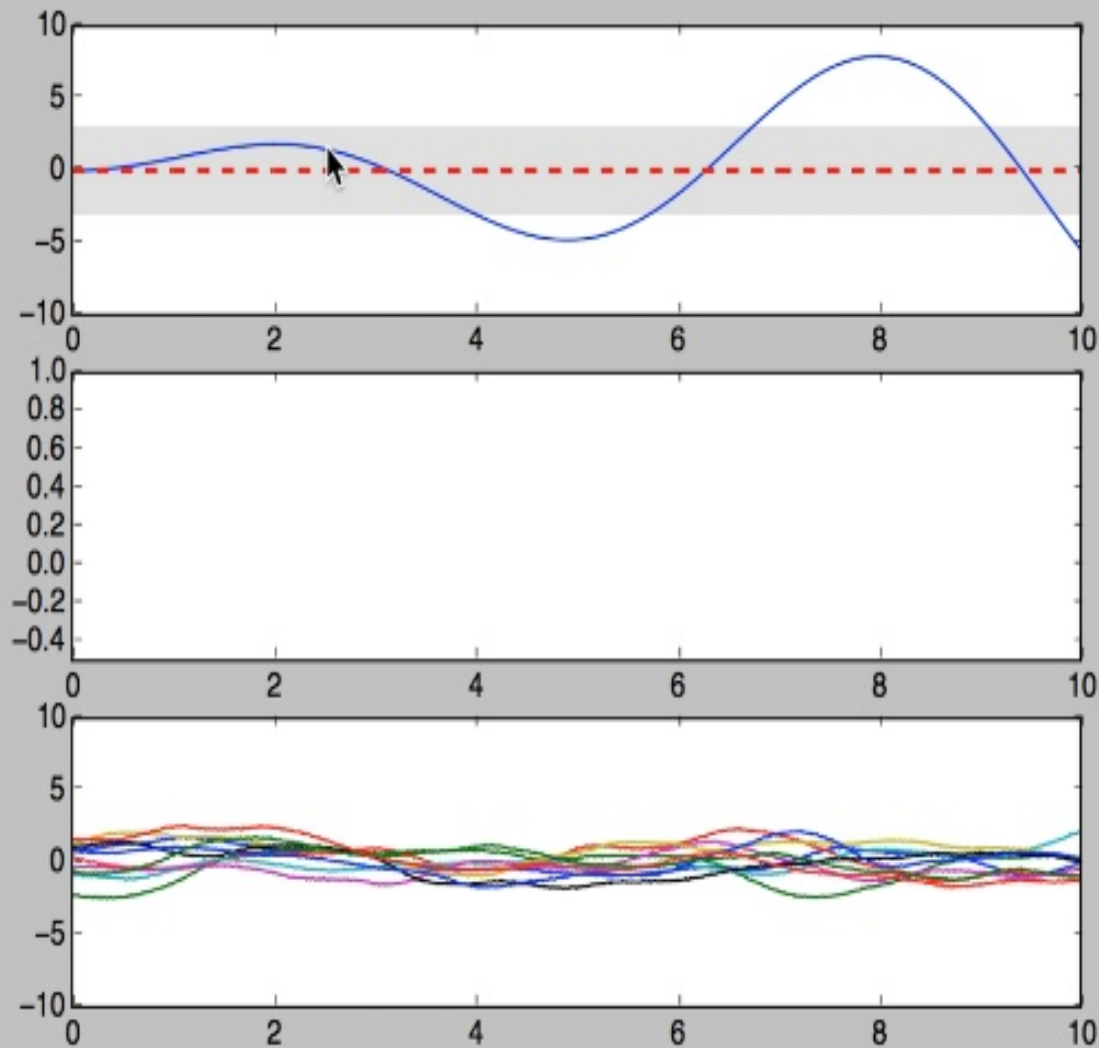
Bayesian Optimization is to *globally* optimize functions that are:

expensive

multi-modal

noisy

blackbox

Bayesian Optimization for "Automating" Data Science on Spark

Automatic Data Science Workflow
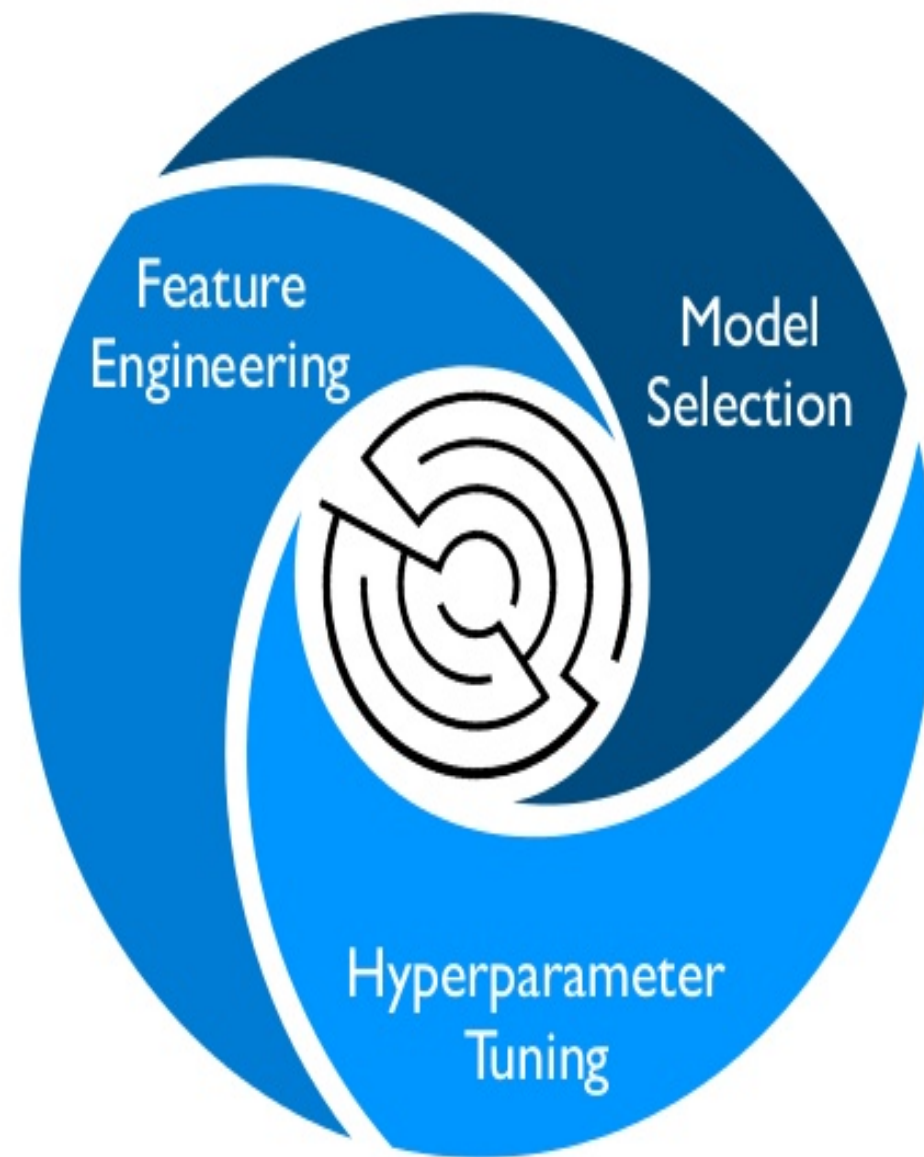
Machines doing part of Data Science

```
┌──────────┐      ┌─────────────────────┐      ┌─────────────────────┐
│          │      │ Feature preprocessing│      │ Feature selection   │
│ Dataset  │─────▶│                     │─────▶│                     │
│          │      │ Data Cleaning       │      │ Dimensionality      │
│          │      │ Generate new features│      │ reduction           │
└──────────┘      └─────────────────────┘      └─────────────────────┘
                                                          │
                         ┌────────────────────┐          │
                         │ Predictive modeling │◀─────────┘     ┌─────────┐
                         │                     │───────────────▶│ Scoring │
                         └────────────────────┘                └─────────┘
```
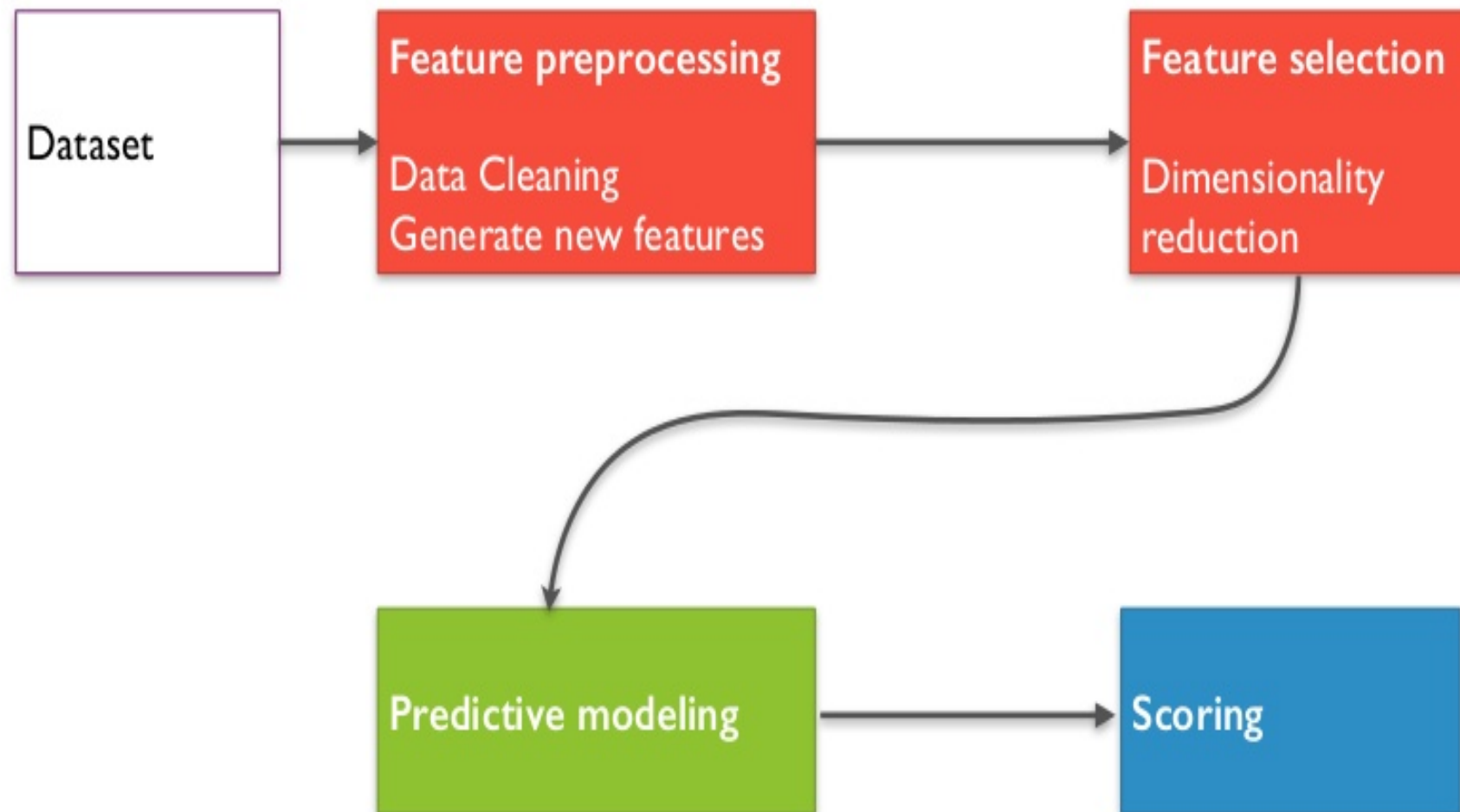
A generic Machine Learning pipeline

Training set

Val/Test set

Feature preprocessing

Truncated SVD

n: # of dims

f-score with target variable

Keep *alpha* percents top features

K-means

k: # of clusters

RandomForest #1

number of trees
maximum depth
percentage of features

...

RandomForest #k

Scoring
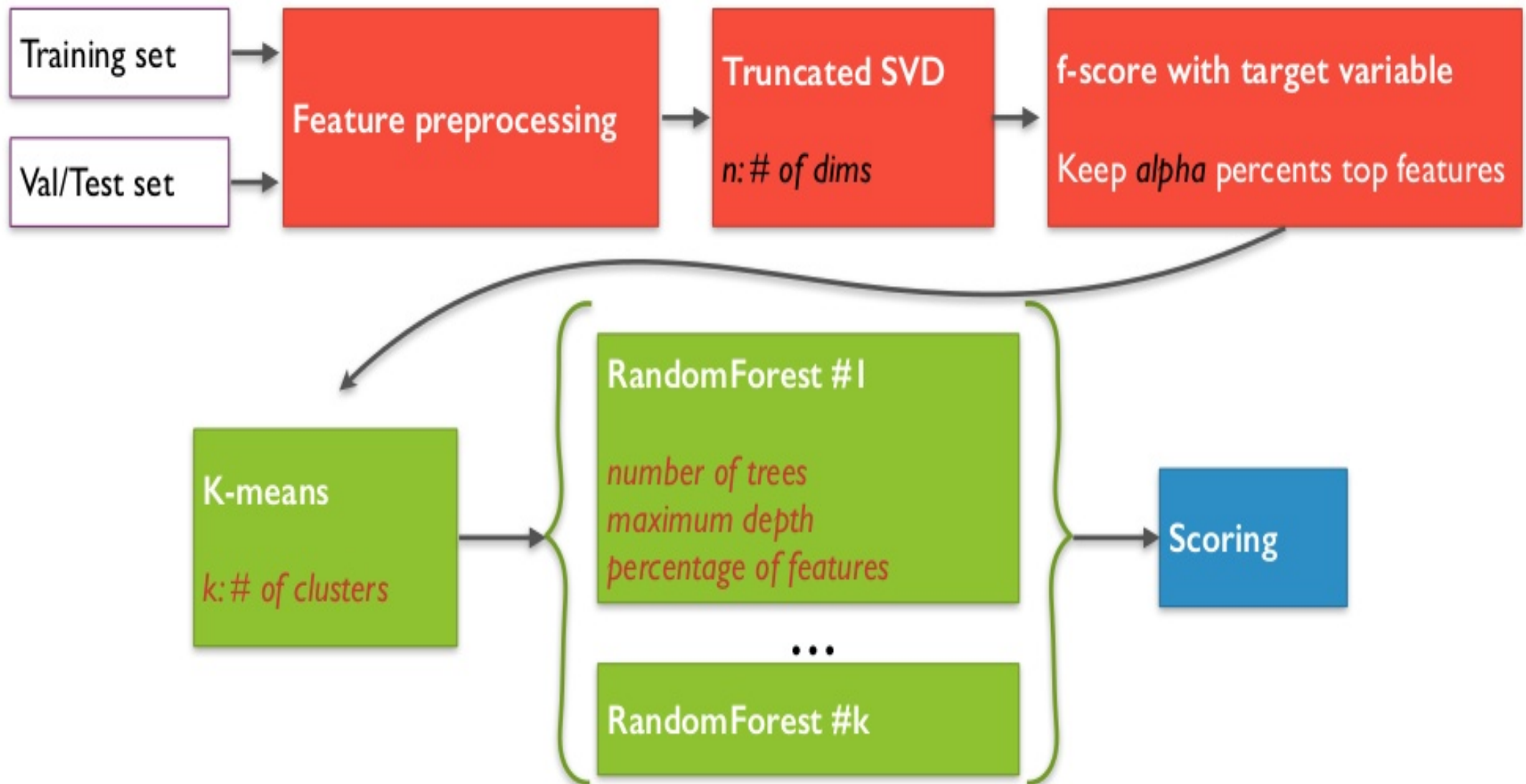
Pipeline on DDF and BayesOpt
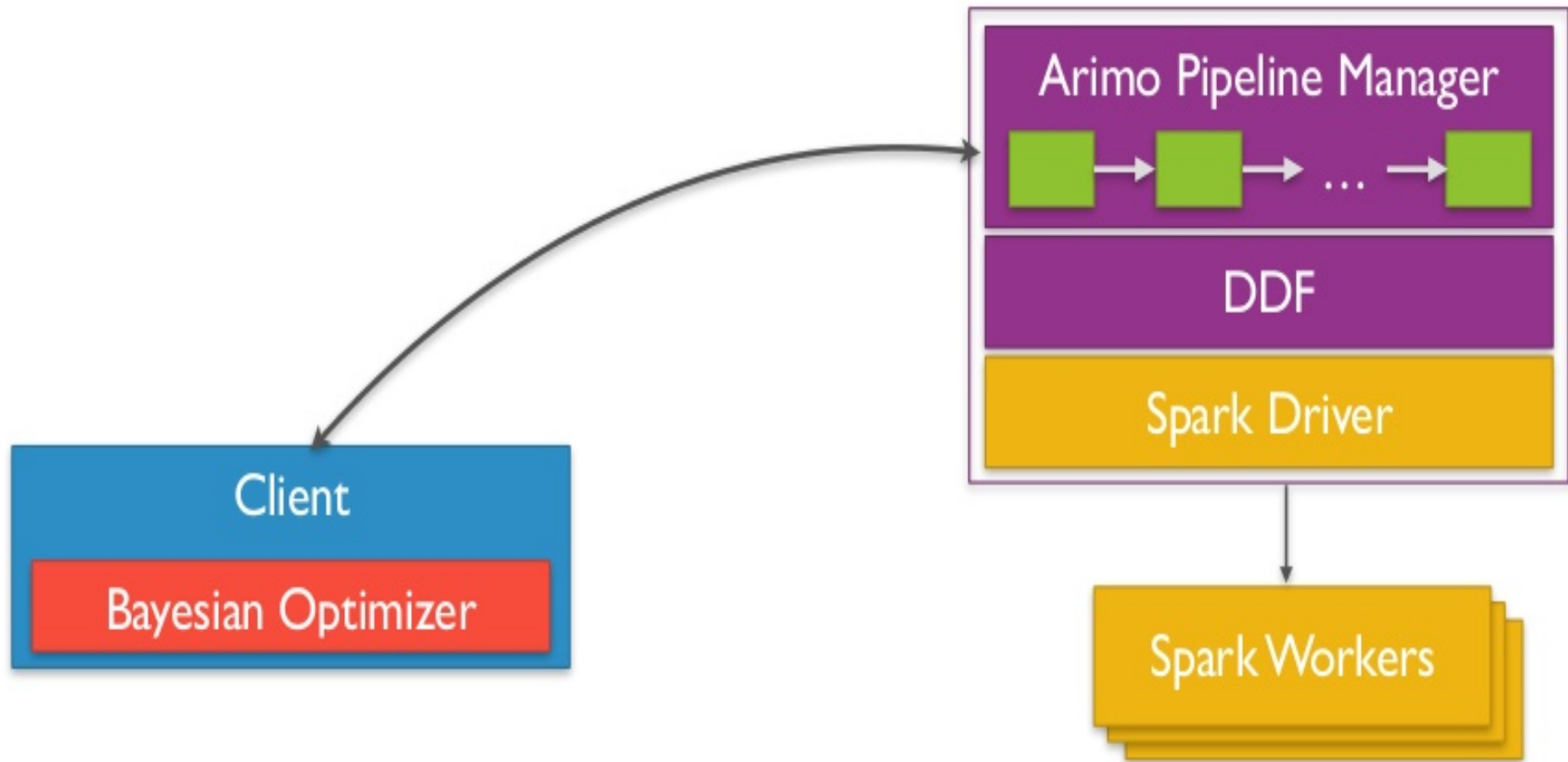
Client uses **Bayesian Optimizer** to select the hyper-parameters of the **pipeline** so that it maximizes the *performance* on a validation set

Pipeline on DDF and BayesOpt

```python
train_ddf = session.get_ddf(…)
valid_ddf = session.get_ddf(…)


optimizer = SpearmintOptimizer(chooser_name='GPEIperSecChooser',
        max_finished_jobs=max_iters, grid_size=5000, ..)


best_params, trace = auto_model(
        optimizer, train_ddf, 'arrdelay',
        classification=True,
        excluded_columns=['actualelapsedtime','arrtime', 'year'],
        validation_ddf=val_ddf)
```

Experimental Results

Experiment 1: SF Crimes

Type of Crime
- ASSAULT
- BURGLARY
- DRUG/NARCOTIC
- FRAUD
- LARCENY/THEFT
- MISSING PERSON
- NON-CRIMINAL
- ROBBERY
- SUSPICIOUS OCC
- VANDALISM
- VEHICLE THEFT
- WARRANTS

Dates	Category	Descript	DayOfWeek	PdDistrict	Resolution	Address	X	Y
2015-05-13 23:53:00	WARRANTS	WARRANT ARREST	Wednesday	NORTHERN	ARREST, BOOKED	OAK ST / LAGUNA ST	-122.4258	37.7745
2015-05-13 23:53:00	OTHER OFFENSES	TRAFFIC VIOLATION ARREST	Wednesday	NORTHERN	ARREST, BOOKED	OAK ST / LAGUNA ST	-122.4258	37.7745
2015-05-13 23:33:00	OTHER OFFENSES	TRAFFIC VIOLATION ARREST	Wednesday	NORTHERN	ARREST, BOOKED	VAN NESS AV / GREENWICH ST	-122.4243	37.8004

Experiment 1: SF Crimes dataset

Hyper-parameter	Type	Range
Number of hidden layers	INT	1, 2, 3
Number of hidden units	INT	64, 128, 256
Dropout at the input layer	FLOAT	[0, 0.5]
Dropout at the hidden layers	FLOAT	[0, 0.75]
Learning rate	FLOAT	[0.01, 0.1]
L2 Weight decay	FLOAT	[0, 0.01]
Logloss on the validation set		
Running time (hours) ~ 40 iterations		

Experiment 1: SF Crimes dataset

Hyper-parameter	Type	Range	Spearmint
Number of hidden layers	INT	1, 2, 3	2
Number of hidden units	INT	64, 128, 256	256
Dropout at the input layer	FLOAT	[0, 0.5]	0.423678
Dropout at the hidden layers	FLOAT	[0, 0.75]	0.091693
Learning rate	FLOAT	[0.01, 0.1]	0.025994
L2 Weight decay	FLOAT	[0, 0.01]	0.00238
Logloss on the validation set			*2.1502*
Running time (hours) ~ 40 iterations			*15.8*

Experiment 1: SF Crimes dataset

2.1600

2.1555   2.1552   2.1551

2.1519

2.1502
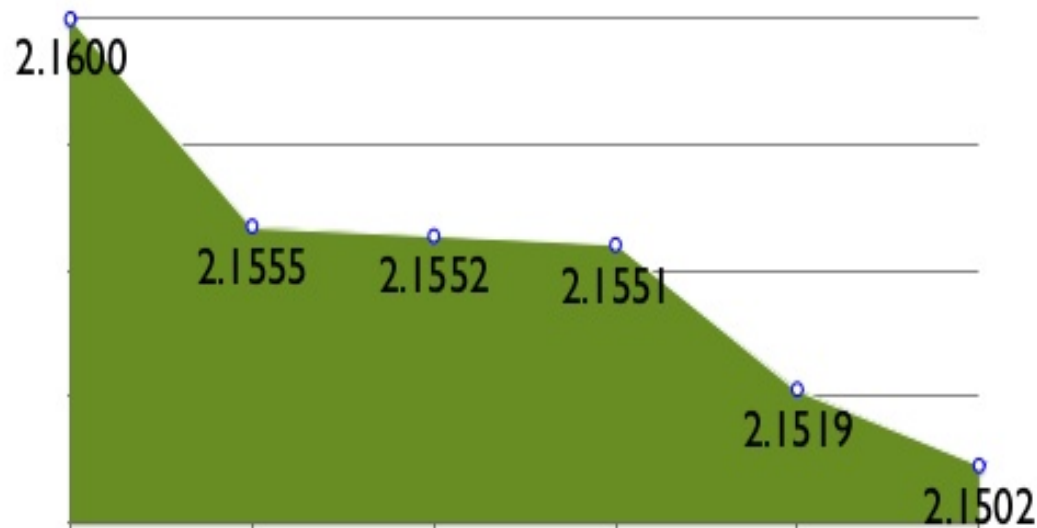
Completed jobs	1	7	16	17	18	39
Elapsed time	1021	8003	12742	1623	1983	31561
Number of layers	1	3	3	2	3	2
Hidden units	64	256	256	256	256	256
Learning rate	0.01	0.1	0.01	0.01	0.021	0.026
Input dropout	0	0.5	0.5	0.463	0.5	0.424
Hidden dropout	0	0	0	0.024	0.089	0.092
Weight decay	0	0	0.002	0.003	0	0.002

Experiment #1: With SigOpt

Hyper-parameter	Type	Range	Spearmint	ΣSIGOPT
Number of hidden layers	INT	1, 2, 3	2	3
Number of hidden units	INT	64, 128, 256	256	256
Dropout at the input layer	FLOAT	[0, 0.5]	0.423678	0.3141
Dropout at the hidden layers	FLOAT	[0, 0.75]	0.091693	0.0944
Learning rate	FLOAT	[0.01, 0.1]	0.025994	0.0979
L2 Weight decay	FLOAT	[0, 0.01]	0.00238	0.0039
Logloss on the validation set			2.1502	2.14892
Running time (hours) ~ 40 iterations			15.8	20.1

SF Crimes - Time to results
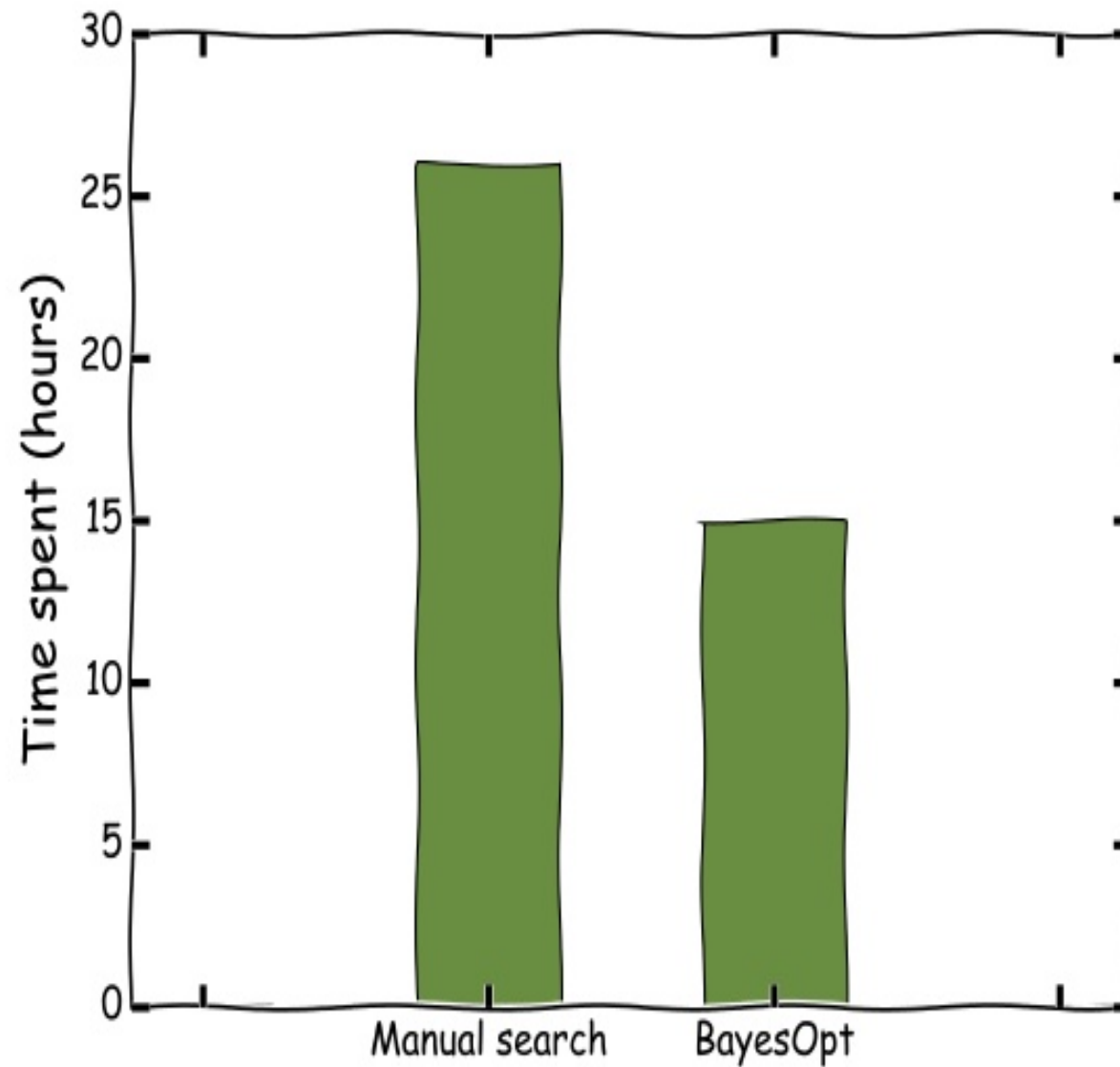
Experiment #2: Airlines data

Year	1987-2008	DepDelay	departure delay, in minutes
Month	1-12	Origin	origin IATA airport code
DayofMonth	1-31	Dest	destination IATA airport code
DayOfWeek	1 (Monday) - 7 (Sunday)	Distance	in miles
DepTime	actual departure time	TaxiIn	taxi in time, in minutes
CRSDepTime	scheduled departure time	TaxiOut	taxi out time in minutes
ArrTime	actual arrival time	Cancelled	was the flight cancelled?
CRSArrTime	scheduled arrival time	CancellationCode	reason for cancellation
UniqueCarrier	unique carrier code	Diverted	1 = yes, 0 = no
FlightNum	flight number	CarrierDelay	in minutes
TailNum	plane tail number	WeatherDelay	in minutes
ActualElapsedTime	in minutes	NASDelay	in minutes
CRSElapsedTime	in minutes	SecurityDelay	in minutes
AirTime	in minutes	LateAircraftDelay	in minutes

Experiment #2

Training set →

Val/Test set →

Feature preprocessing

~900 features

→

Truncated SVD

n: # of dims

→

f-score with target variable

Keep *alpha* percents top features

K-means

k: # of clusters

→

RandomForest #1

number of trees
maximum depth
percentage of features

...

RandomForest #k

→ **Scoring**
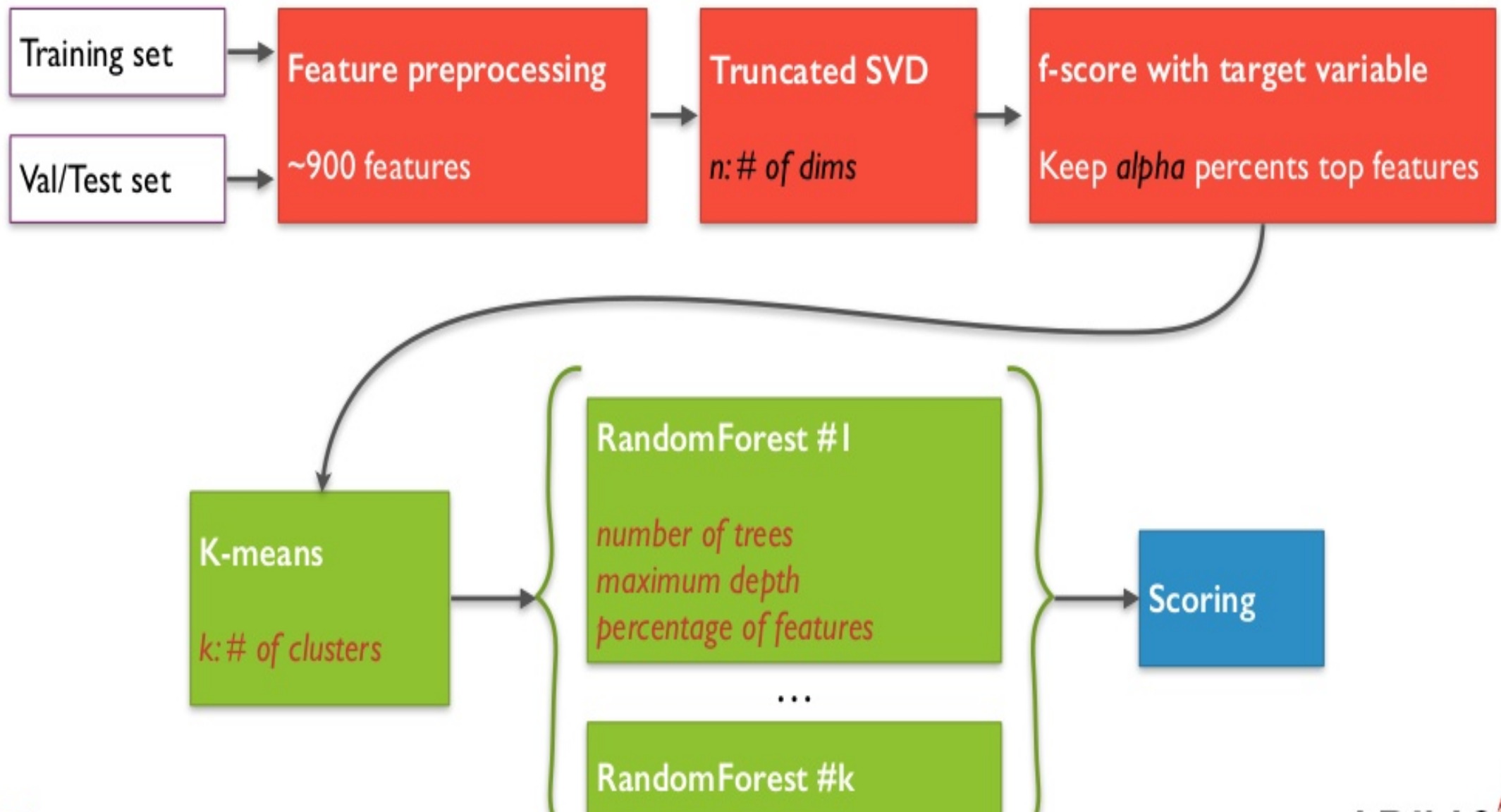
Experiment #2: Hyper-parameters

Hyperparameter	Type	Range	BayesOpt
Number of SVD dimensions	INT	[5, 100]	98
Top feature percentage	FLOAT	[0.1, 1]	0.8258
k (# of clusters)	INT	[1, 6]	2
Number of trees (RF)	INT	[50, 500]	327
Max. depth (RF)	INT	[1, 20]	12
Min. instances per node (RF)	INT	[1, 1000]	414
F1-score on validation set			*0.8736*

Summary

1. Bayesian Optimization for Hyper-parameter Tuning

2. Bayesian Optimization for

   "Automating" Data Science on Spark

3. Experiments

Getting Started

- Blogpost: http://goo.gl/PFyBKl

- Open-source: spearmint, hyperopt, SMAC, AutoML

- Commercial: Whetlab, SigOpt, ...