Machine Learning for Data Analysis

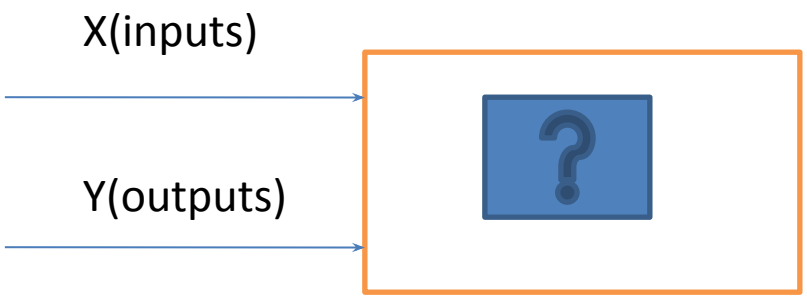
Ravitha Rajalakshmi N
Asst Professor
Dept of IT, PSG CT

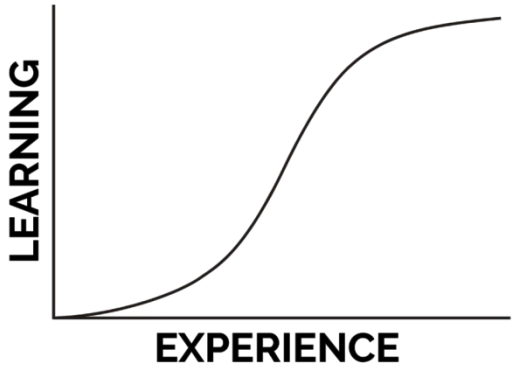
Outline

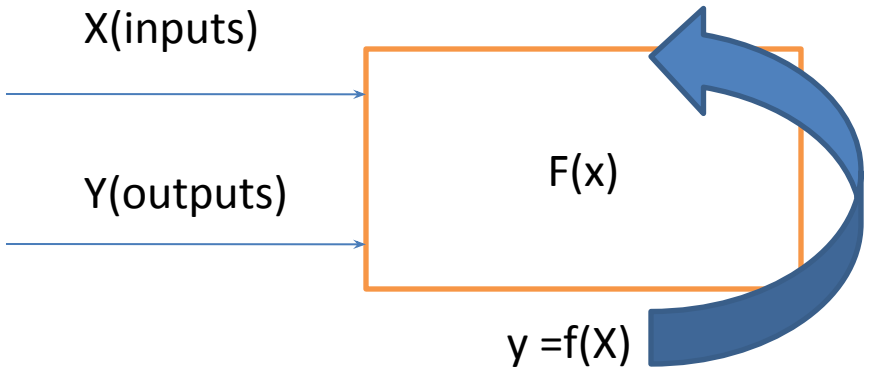
- Introduction
- Machine Learning Workflow
- Case Study : Normal ML pipeline
- Difficulties (or) Challenges
- Hyperparameter Optimization
- AutoML
- Case Study : AutoML
- Research Challenges

Learning

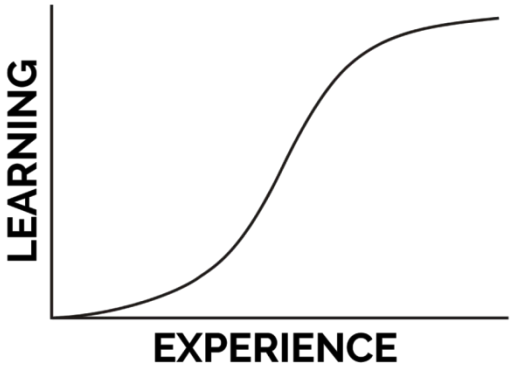
A computer program is said to learn from Experience (Dataset) E with some class of Tasks T and Performance measure P if its performance at Tasks in T measured by P , improves with experience E.





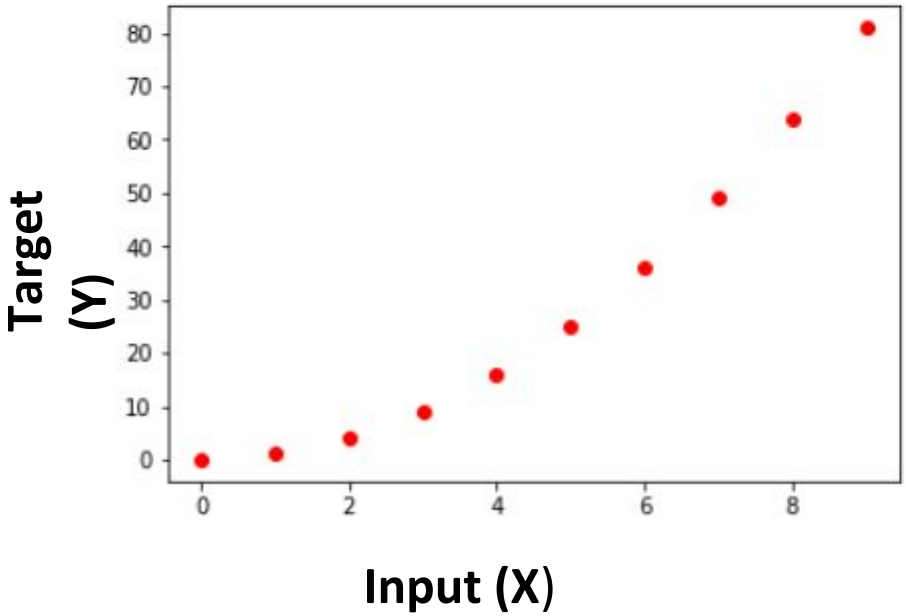


Iterative Process



Regression : MSE
Classification : Binary_ cross entropy

Model family of functions



Y= sin(x)

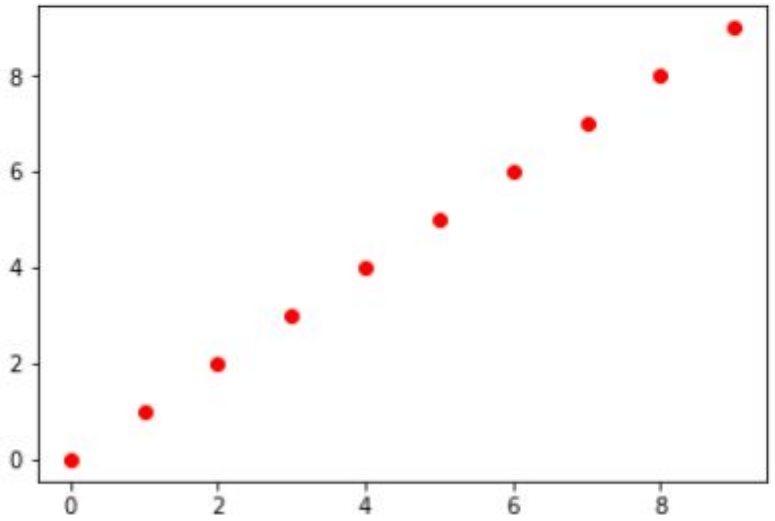
Y = 4* x +!2

Y = 3*x + 8

Y= X^2

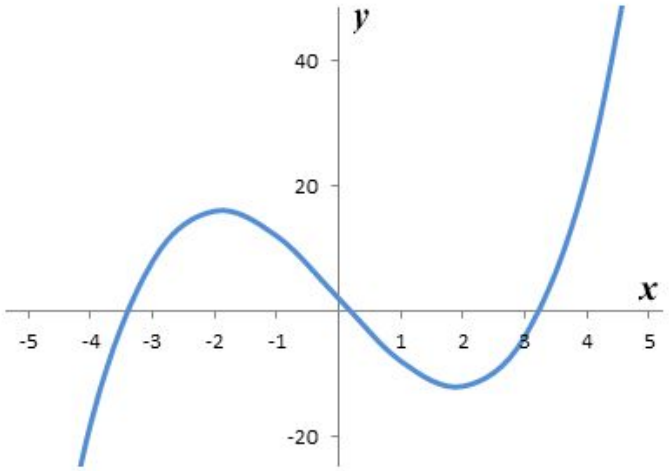
Can ML learn any function between input and output?

Linear Function



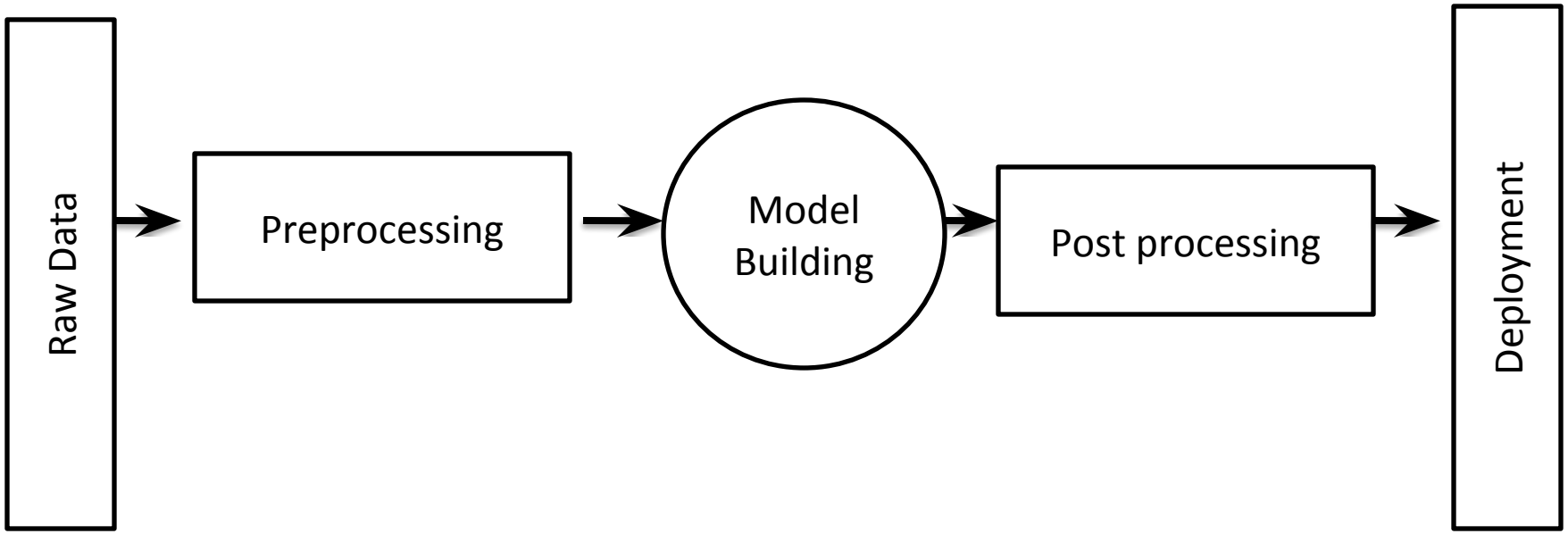
Linear Regression
SVM

Non - Linear Function

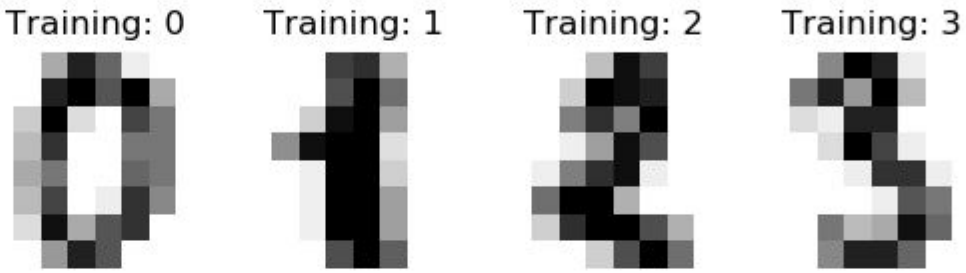


NN
SVM with RBF

Machine Learning Workflow



MNIST Dataset



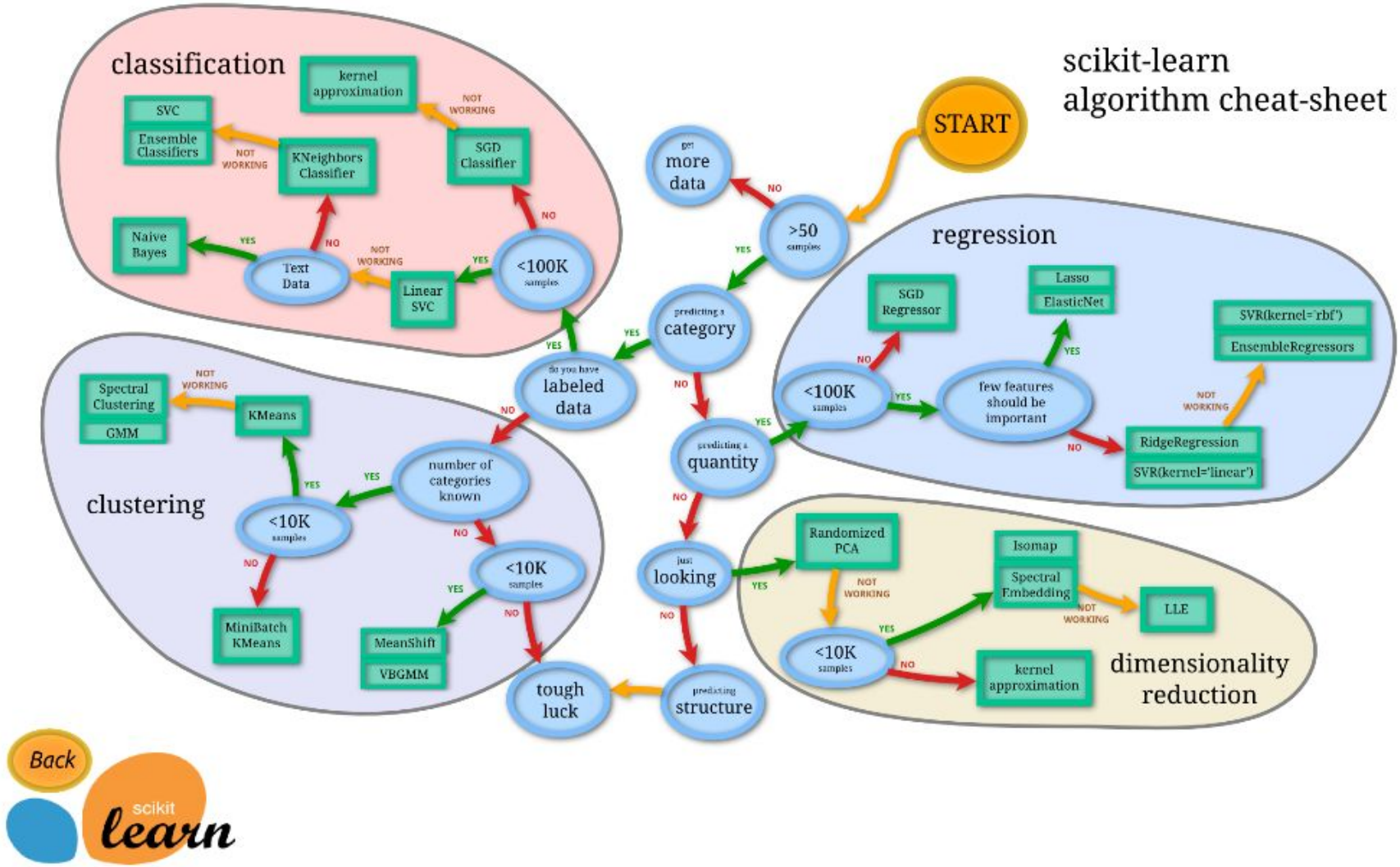
28 * 28

```
000 001 002 003 ... 026 027
028 029 030 031 ... 054 055
056 057 058 059 ... 082 083
 |   |   |   |  ...  |   |
728 729 730 731 ... 754 755
756 757 758 759 ... 782 783
```

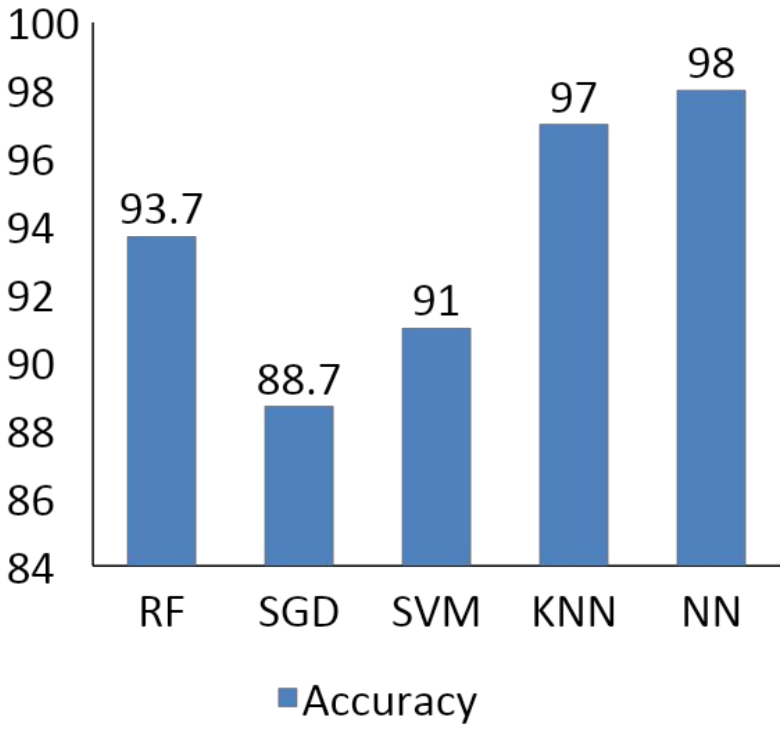
X

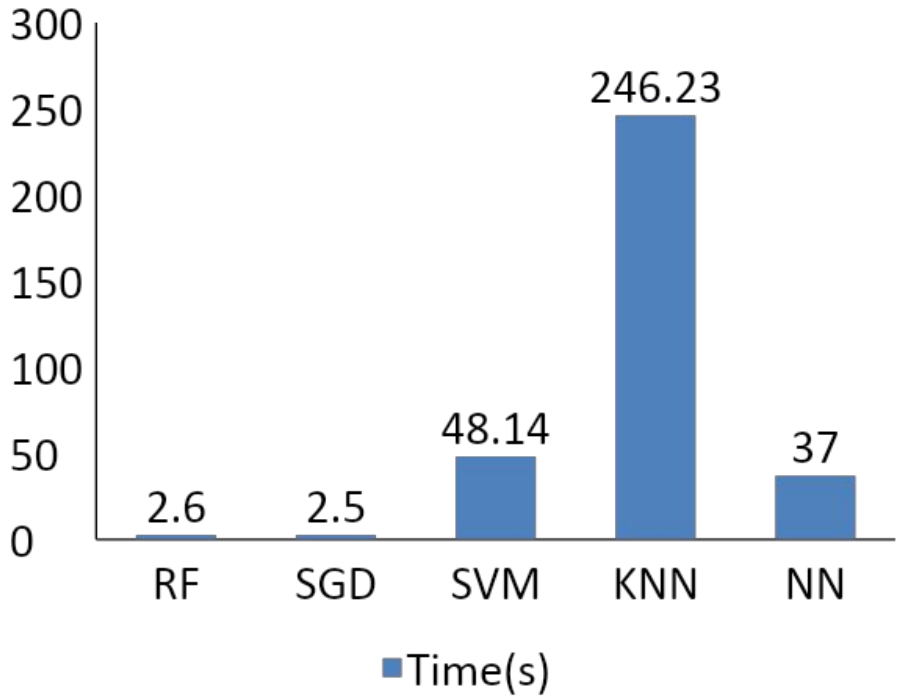
784 * 1

Target
(Y)


scikit-learn
algorithm cheat-sheet
START
get more data
NO
>50 samples
YES
predicting a category
YES
do you have labeled data
YES
NO
predicting a quantity
YES
NO
just looking
YES
NO
predicting structure
tough luck
classification
kernel approximation
NOT WORKING
SVC
Ensemble Classifiers
NOT WORKING
KNeighbors Classifier
SGD Classifier
NO
Naive Bayes
YES
Text Data
NO
NOT WORKING
Linear SVC
YES
<100K samples
NO
clustering
Spectral Clustering
GMM
NOT WORKING
KMeans
YES
number of categories known
YES
<10K samples
NO
MiniBatch KMeans
NO
<10K samples
YES
MeanShift
VBGMM
NO
regression
SGD Regressor
NO
<100K samples
YES
few features should be important
YES
Lasso
ElasticNet
NO
RidgeRegression
SVR(kernel='linear')
NOT WORKING
SVR(kernel='rbf')
EnsembleRegressors
dimensionality reduction
Randomized PCA
YES
NOT WORKING
<10K samples
YES
Isomap
Spectral Embedding
NOT WORKING
LLE
NO
kernel approximation
Back
scikit learn

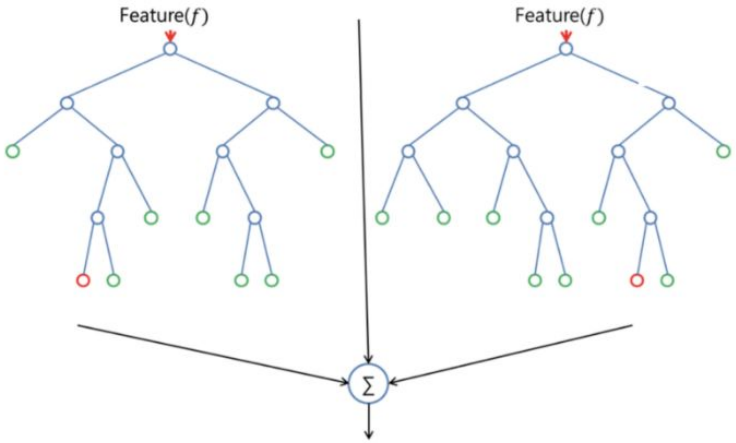
Algorithm and Model Comparison





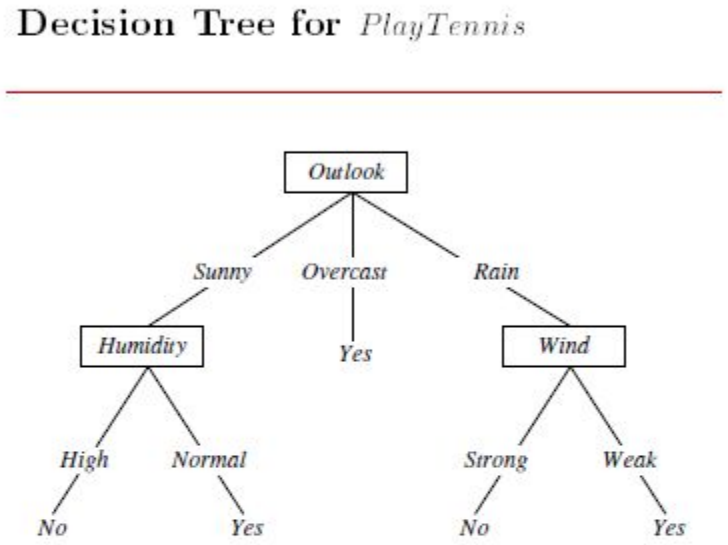
Random Forest

Ensemble of multiple decision trees created using random set of features and random set of training examples



Sample Decision Tree

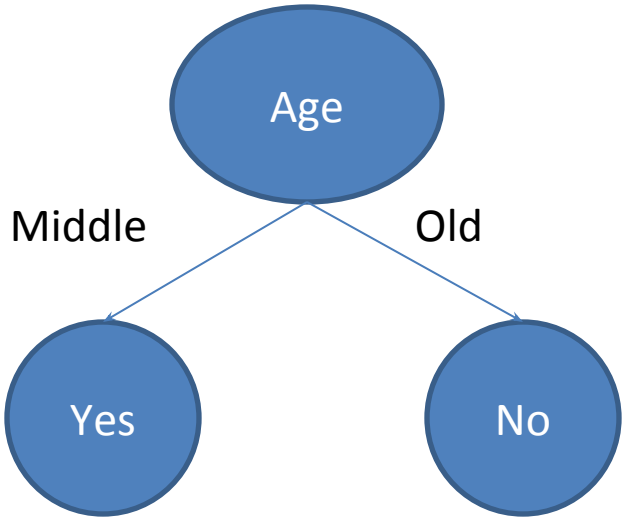
Day	Outlook	Temperature	Humidity	Wind	PlayTenni
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No



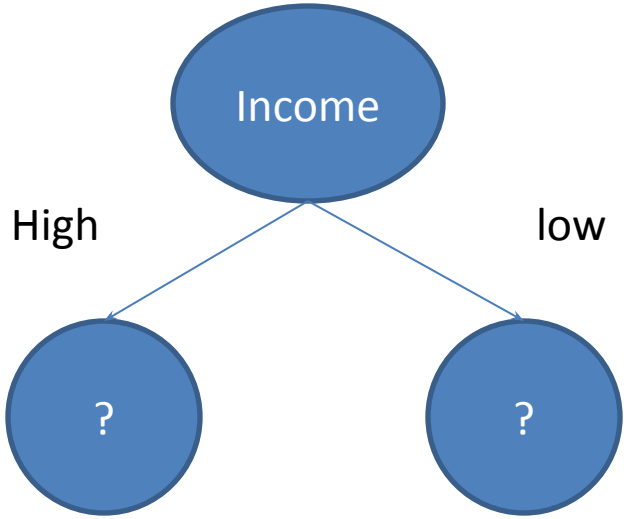
How attributes are selected ?

- Using Metrics
  - Information Gain
  - Entropy

Age	Income	Label
Middle	High	Yes
Middle	Low	Yes
Old	High	No
Old	Low	No



Entropy = 0



Entropy = 1

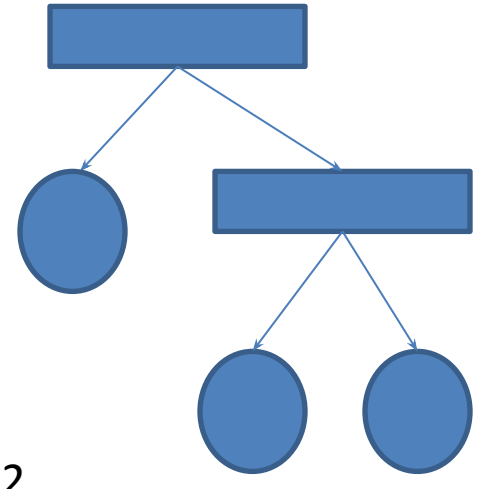
Random Forest classifier

Day	Outlook	Temperature	Humidity	Wind	PlayTenni
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

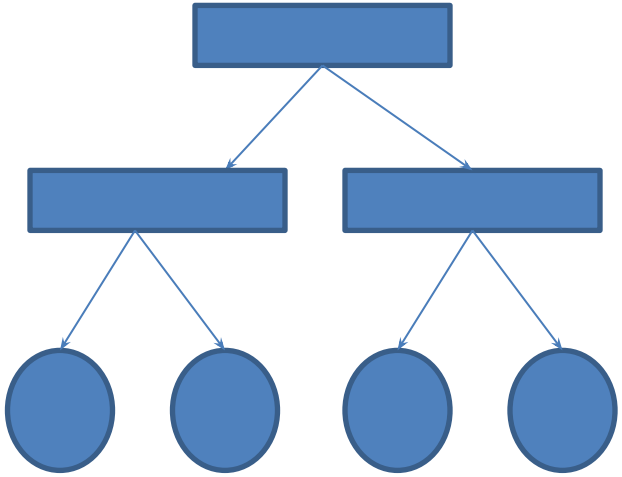
Outlook	Temperature	Play_Tennis
Tuple1		
Tuple 4		
Tuple 8		

Temperature	Wind	Play_Tennis
Tuple 2		
Tuple 5		
Tuple 8		

Tree 1



Tree 2



Parameters vs Hyper-parameters

- Hyper-parameters need to be initialized before training a model.
- ***Model parameters*** are the properties of training data that will learn on its own during training by the classifier or other ML model.
  - Split points in Decision Tree

Parameter Selection is Intuitive

Well Crafted algorithms to choose parameters

For all untried features:

1. Compute **Entropy**
2. Choose the one with minimum entropy
3. Split the dataset based on feature
4. Repeat steps 2 - 3

Minimize difference between actual and predicted

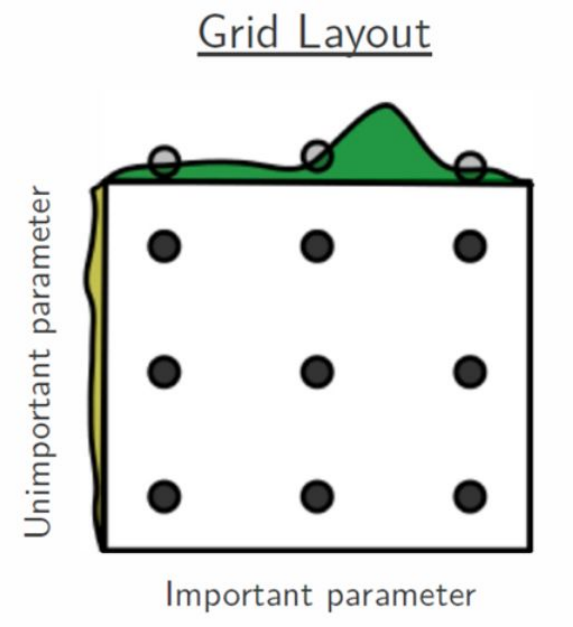
Objective function is known

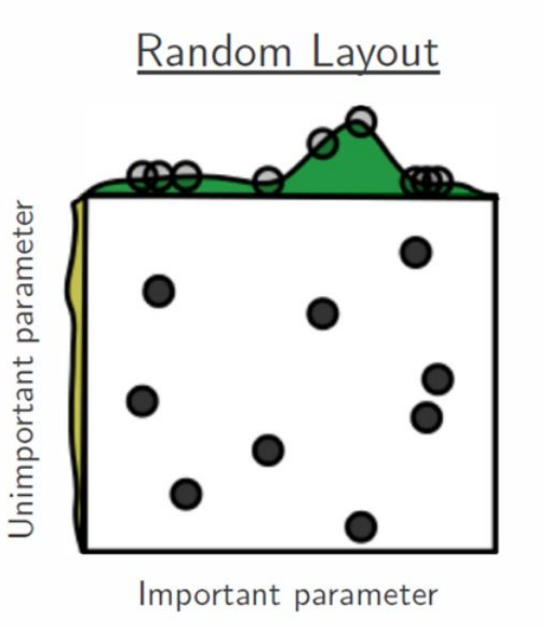
Hyper-parameter Selection is non - intuitive

Objective function is unknown

Hyper parameter Optimization

- Grid Search (Try all Configs) Boredom!!
- Random Search (Try your luck !!)



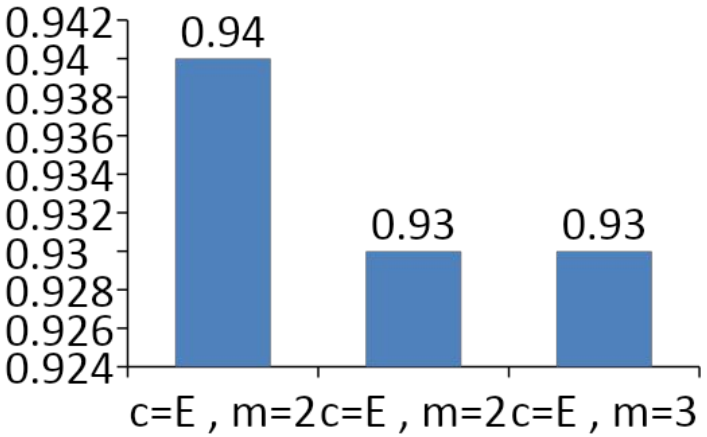


Hyperparameter Optimization

Grid Search

Min_sample_Split / Metric	2	10
Entropy	0.954	0.953
Information Gain	0.953	<0.953

Random Search



Hyperparameters

1. n_estimators
2. max_features
3. n_jobs
4. random_state
5. min_sample_split

Is there a different approach?

Is there an intelligent way of doing things??

What will a human do if task is unknown??

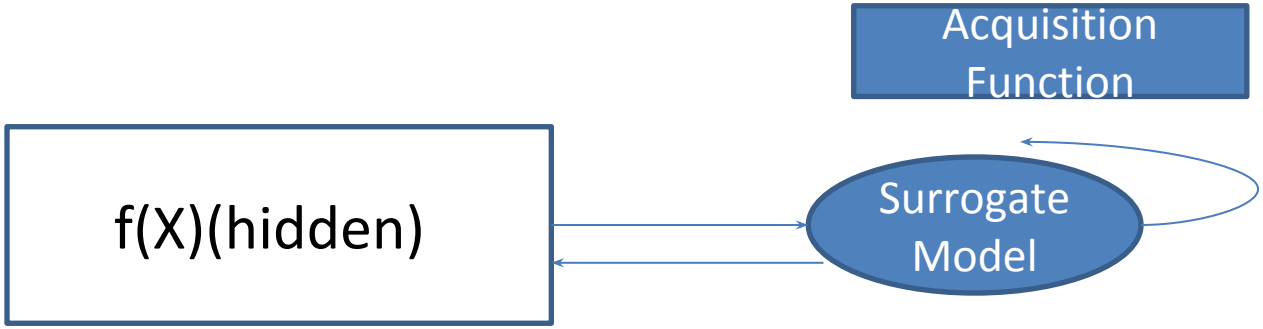
Use the past experience

AutoML

- Targets on the progressive optimization of machine learning
- ML for non ML experts
- It is used for
  - Analyse the importance of hyper-parameters
  - Development of software packages which can be instantiated in a data driven way.

Sequential Model based Optimization

- Bayesian Optimization
- Tree structured parzen estimator

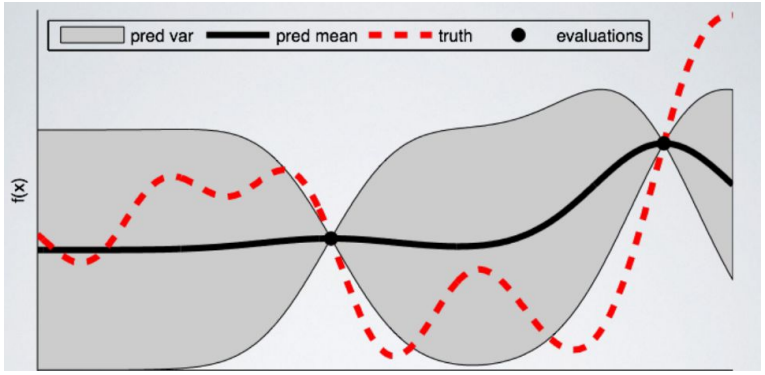


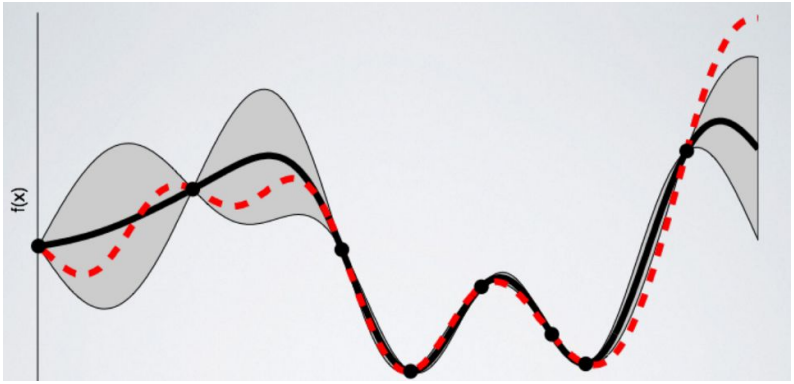
Methods of AutoML

- Bayesian Optimization
- Meta Learning
- Transfer Learning
- Combinatorial Optimization

Bayesian Optimization

$$P(\,score \mid hyperparameters)$$

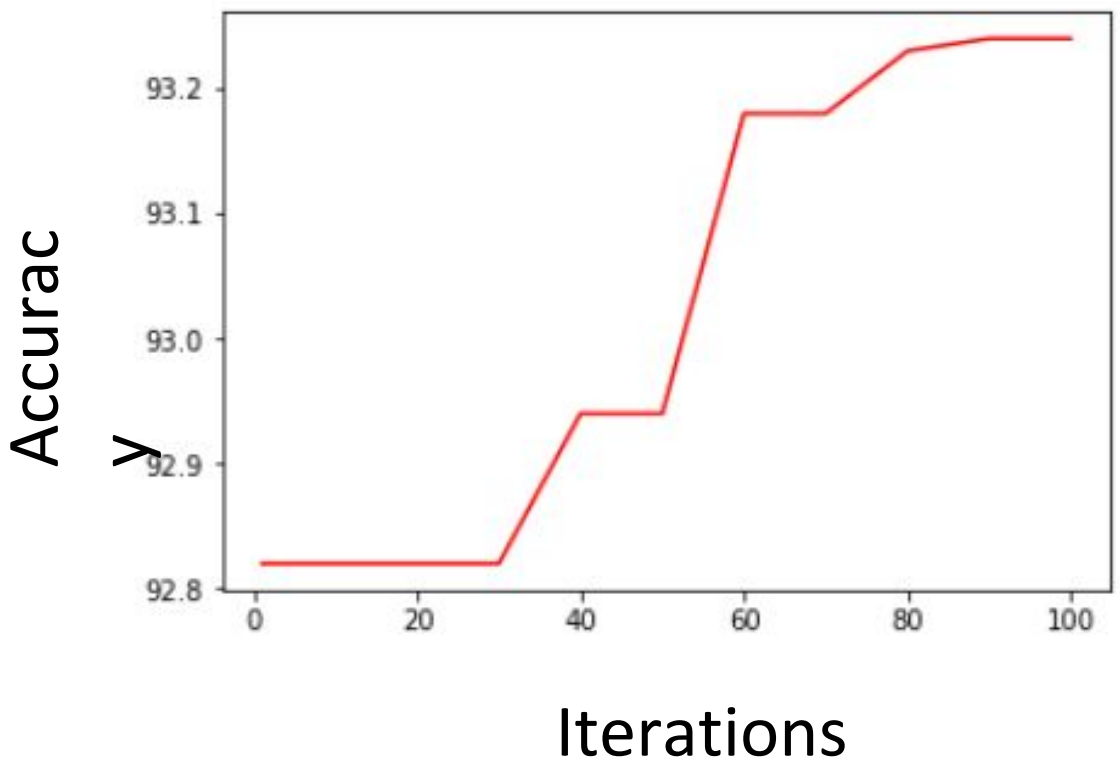


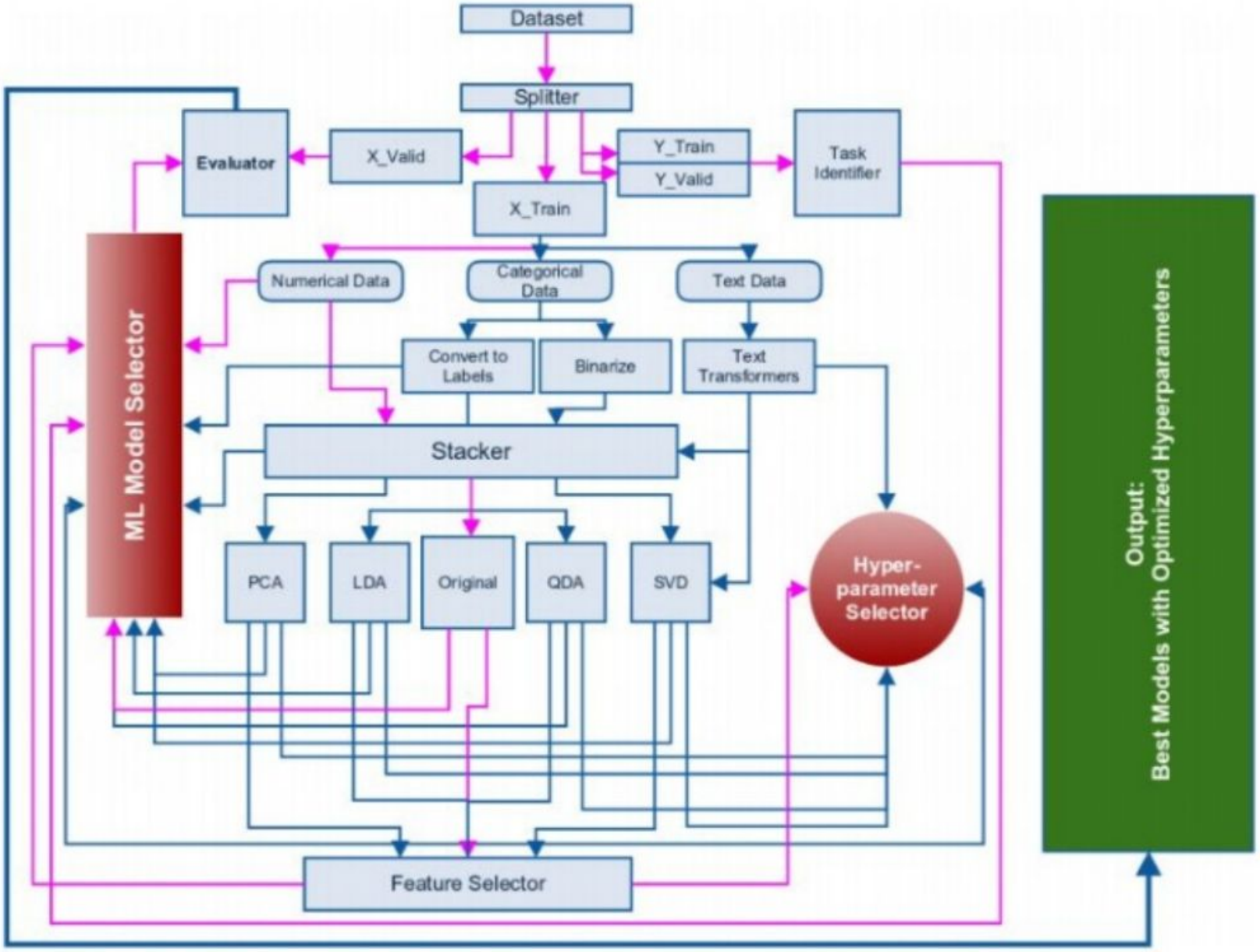


Popular Automated Machine Learning Tools

- Auto-sklearn
- TPOT
- Hyperopt
- Auto-WEKA
- Spearmint

MNIST Optimization using Hyperopt

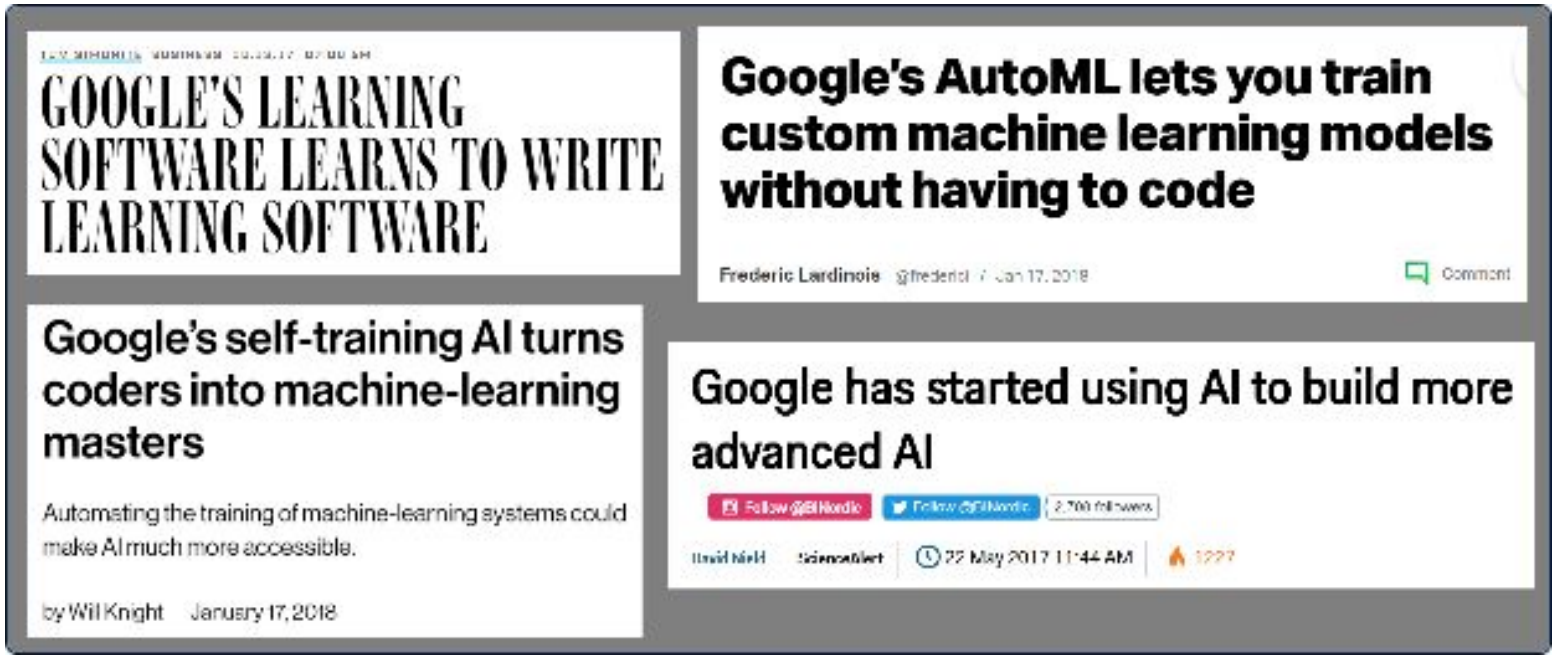



Dataset
Splitter
Evaluator
X_Valid
Y_Train
Y_Valid
Task Identifier
X_Train
ML Model Selector
Numerical Data
Categorical Data
Text Data
Convert to Labels
Binarize
Text Transformers
Stacker
PCA
LDA
Original
QDA
SVD
Hyper-parameter Selector
Feature Selector
Output:
Best Models with Optimized Hyperparameters

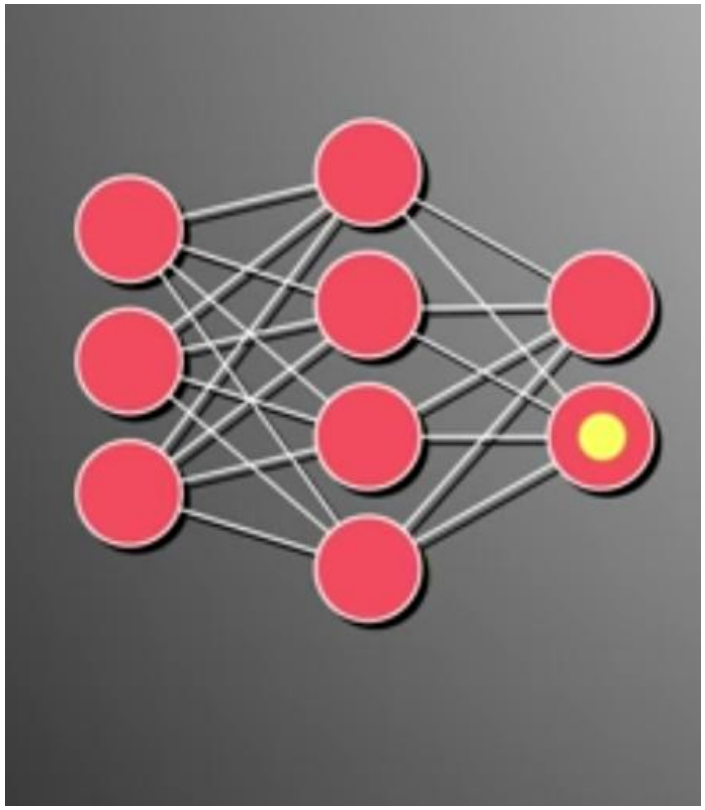
The basic pipeline of every AutoML framework:

- Data Pre-processing
  - Converting the data to tabular form.
  - Splitting the test, train and validation data.
- Feature Engineering
  - Label or one hot encoders for categorical variables.
  - TF-IDF or Bag Of Words for text variables.
- Feature Stacking
  - Combining different features
- Decomposition
  - For high dimension data PCA is used.
  - For text data - SVD is applied after converting text to sparse matrix.
- Feature Selection
  - Greedy Forward Selection
  - Greedy backward elimination
  - Using models like LASSO or Random Forest for implicit selection.
- Model selection and Hyper Parameter tuning
  - Grid Search
  - Random Search
  - Bayesian Search
- Evaluation of model

Neural Architecture Search



Neural Network



Y= sigmoid(W2, sigmoid (W1,X))

Neural Architecture Search

"Designing neural nets is extremely time intensive, and requires an expertise that limits its use to a smaller community of scientists and engineers. That's why we've created an approach called AutoML, showing that ***it's possible for neural nets to design neural nets****."*

-Google CEO

Basic Idea

- A neural network architecture can be described in a few parameters: The number of layers, the number of nodes.
- For a convolutional neural network (CNN), this is the number of filters and filter size.

FLOWCHART

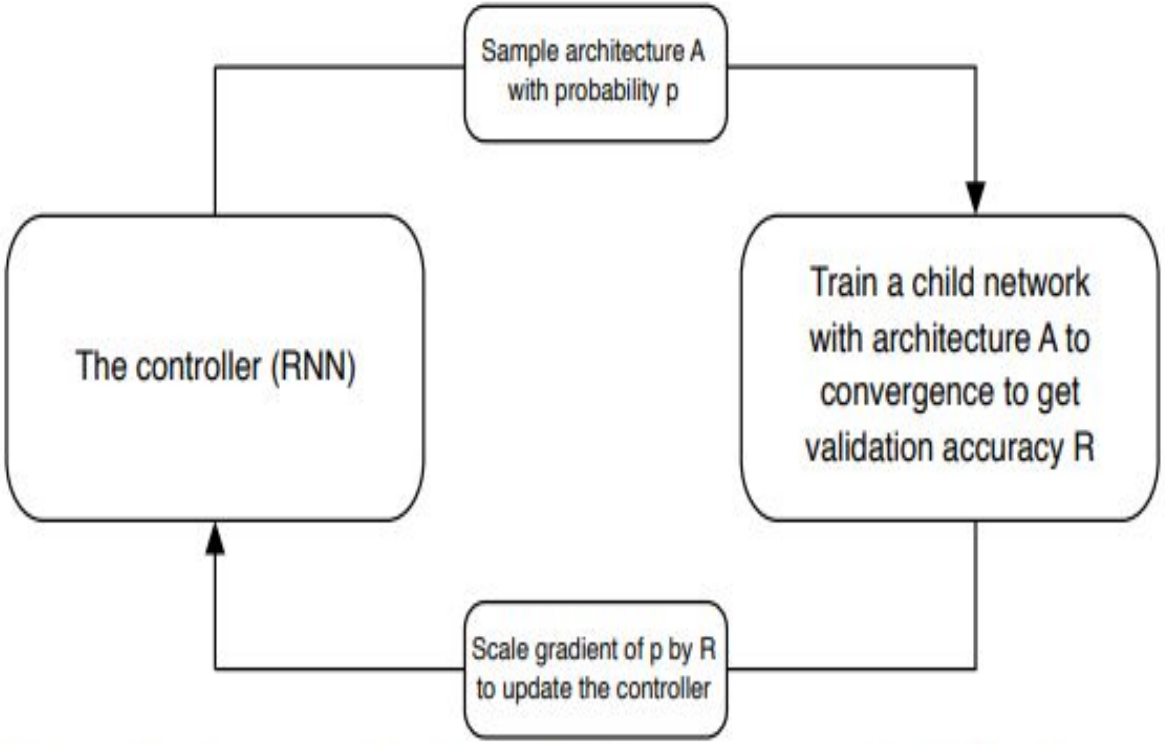
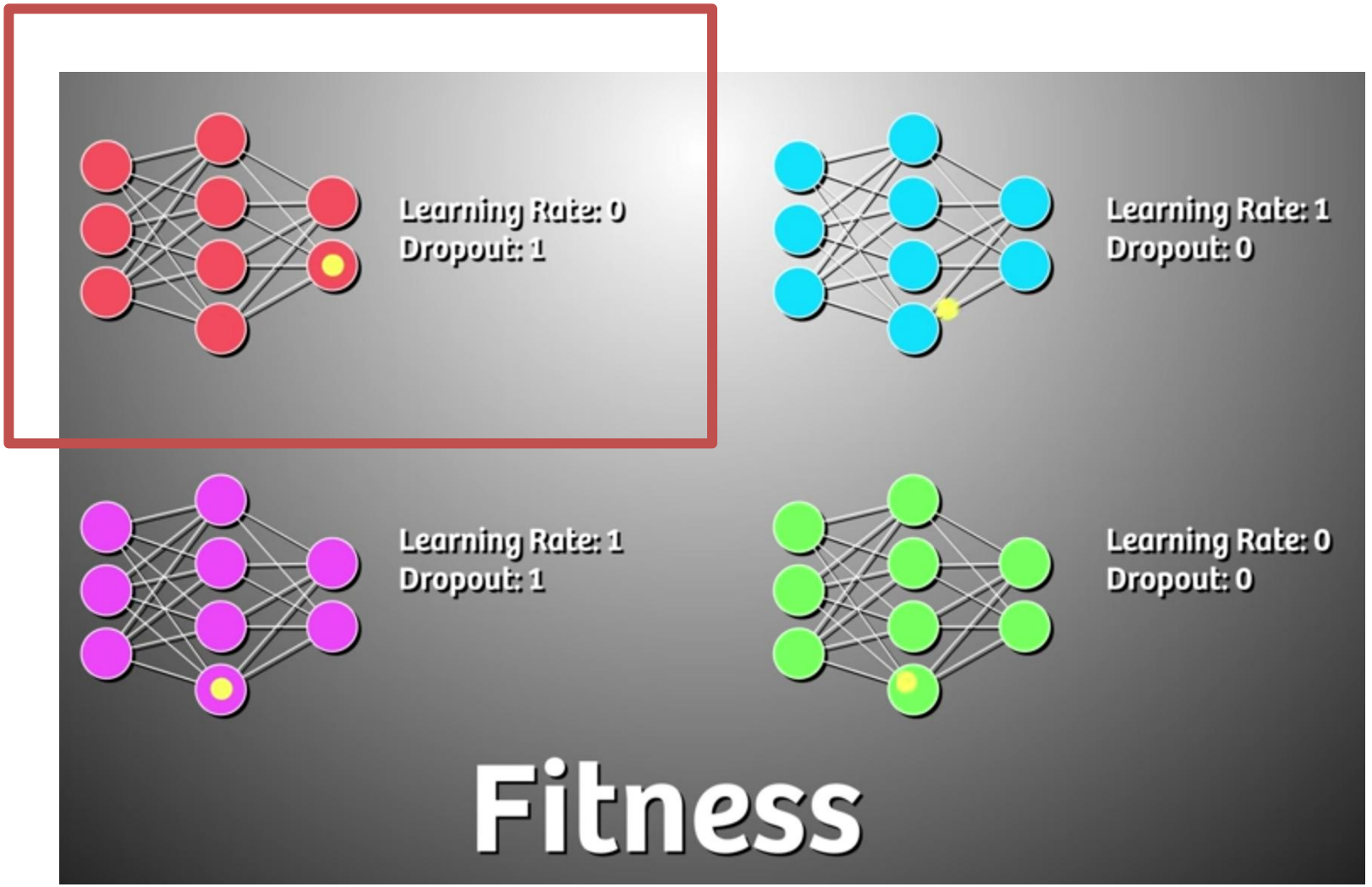
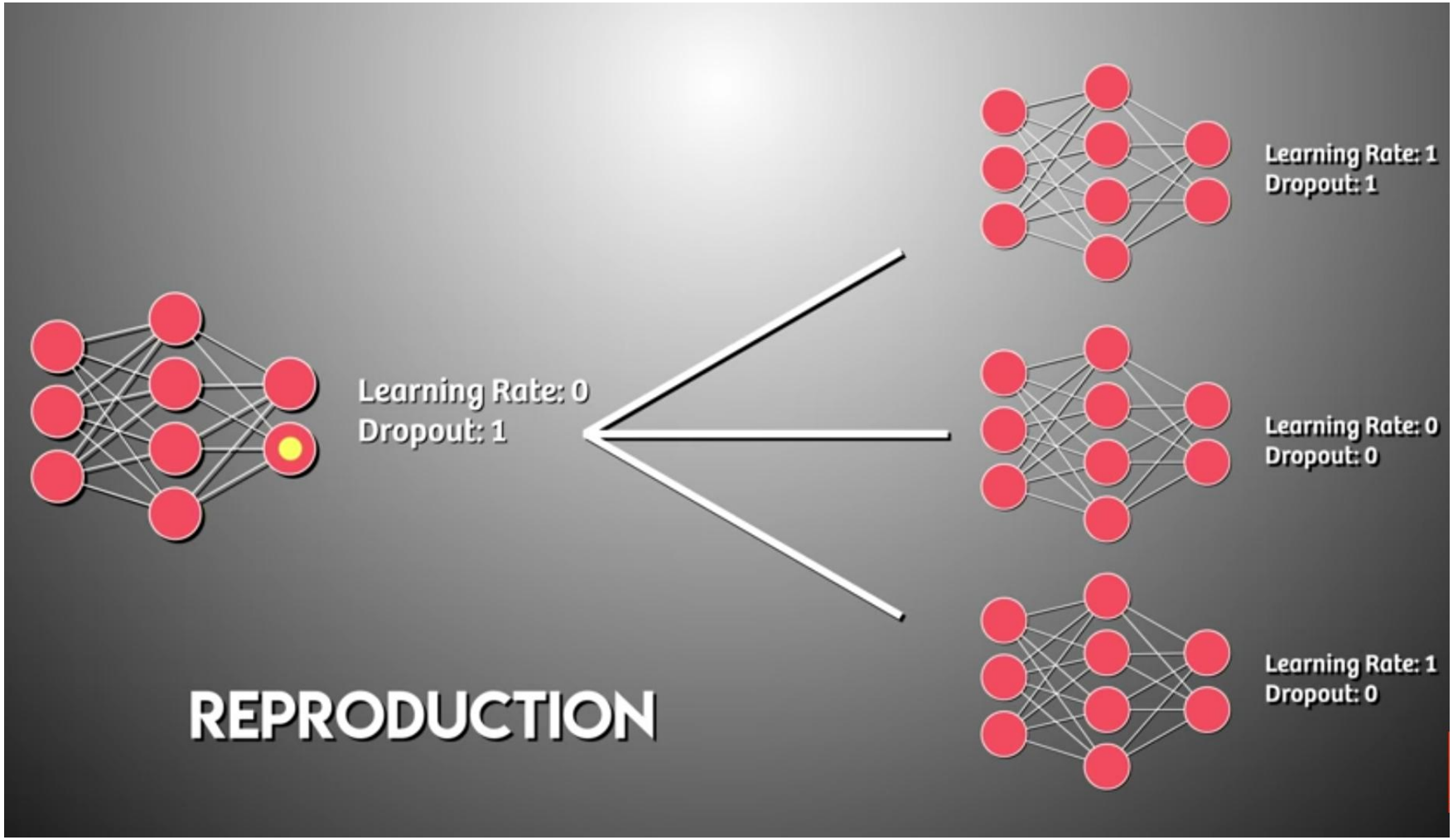


Figure 1. Overview of Neural Architecture Search [71]. A controller RNN predicts architecture A from a search space with probability p. A child network with architecture A is trained to convergence achieving accuracy R. Scale the gradients of p by R to update the RNN controller.


Learning Rate: 0
Dropout: 1
Learning Rate: 1
Dropout: 0
Learning Rate: 1
Dropout: 1
Learning Rate: 0
Dropout: 0
Fitness


Learning Rate: 0
Dropout: 1
Learning Rate: 1
Dropout: 1
Learning Rate: 0
Dropout: 0
Learning Rate: 1
Dropout: 0
REPRODUCTION

Tools

- Auto keras **(Open Source)**
- AutoML (Auto Vision API in Google Cloud Platform)

Drawbacks on AutoML

- Works with pre-processed tabular / Image Data
- Limited support for feature selection

Content Overload !!

- https://www.ml4aad.org/
- http://simonwenkel.com/
- Siraj Raval Video on “AI that Creates AI”

- Acknowlegement : Onepanel for free GPU Credits

Thanks for your Support

Drop in your valuable feedback at nrr.it@psgtech.ac.in
Suggestions are welcome!!