Practice of AI

C2: Machine learning & Data analyze

Jim Xie

2020/3/6

Symbols

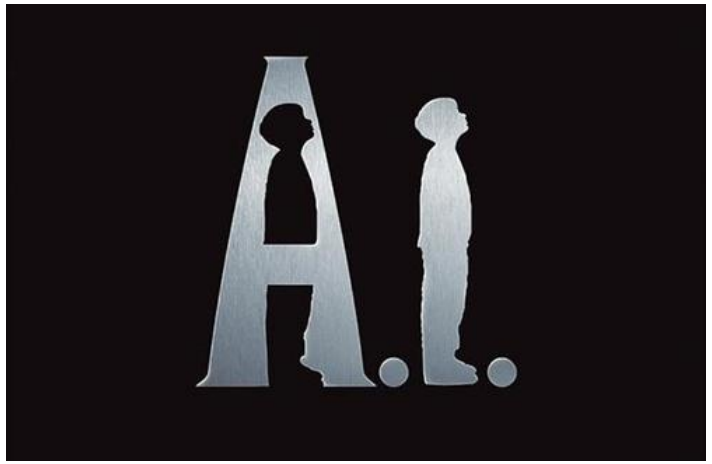
σ ω δ ε

ξ ψ θ γ

Outline

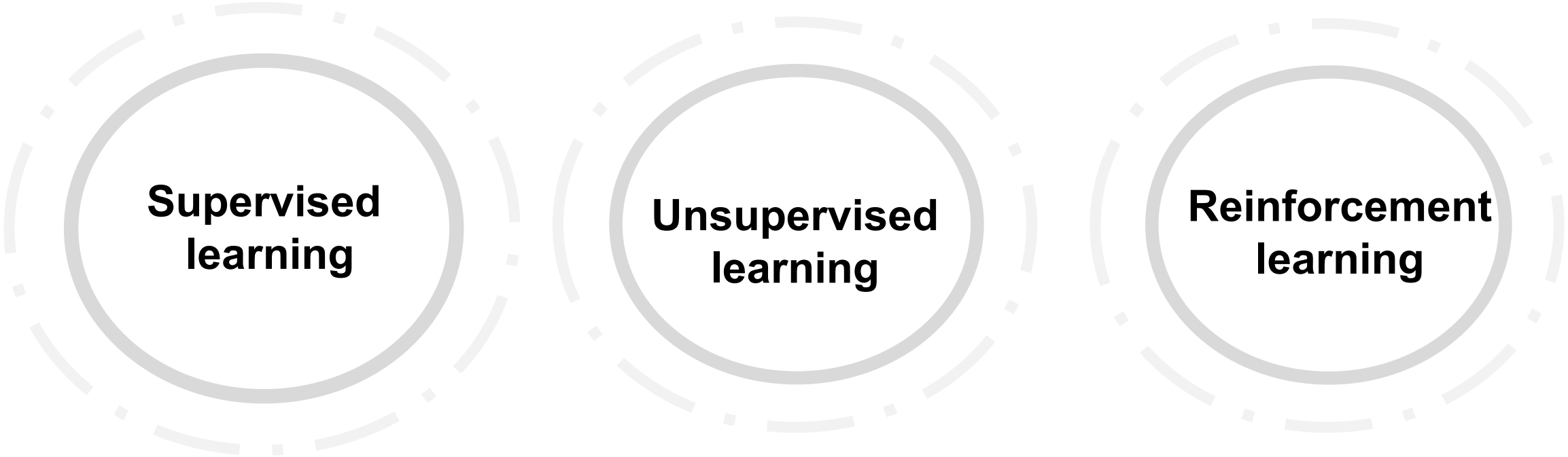
1. Goal
2. ML workflow introduction
3. Basic math knowledge introduction
4. Time series forecasting demo
5. Brief summary

Goal



Getting start for data analyze with ML

Category by sample



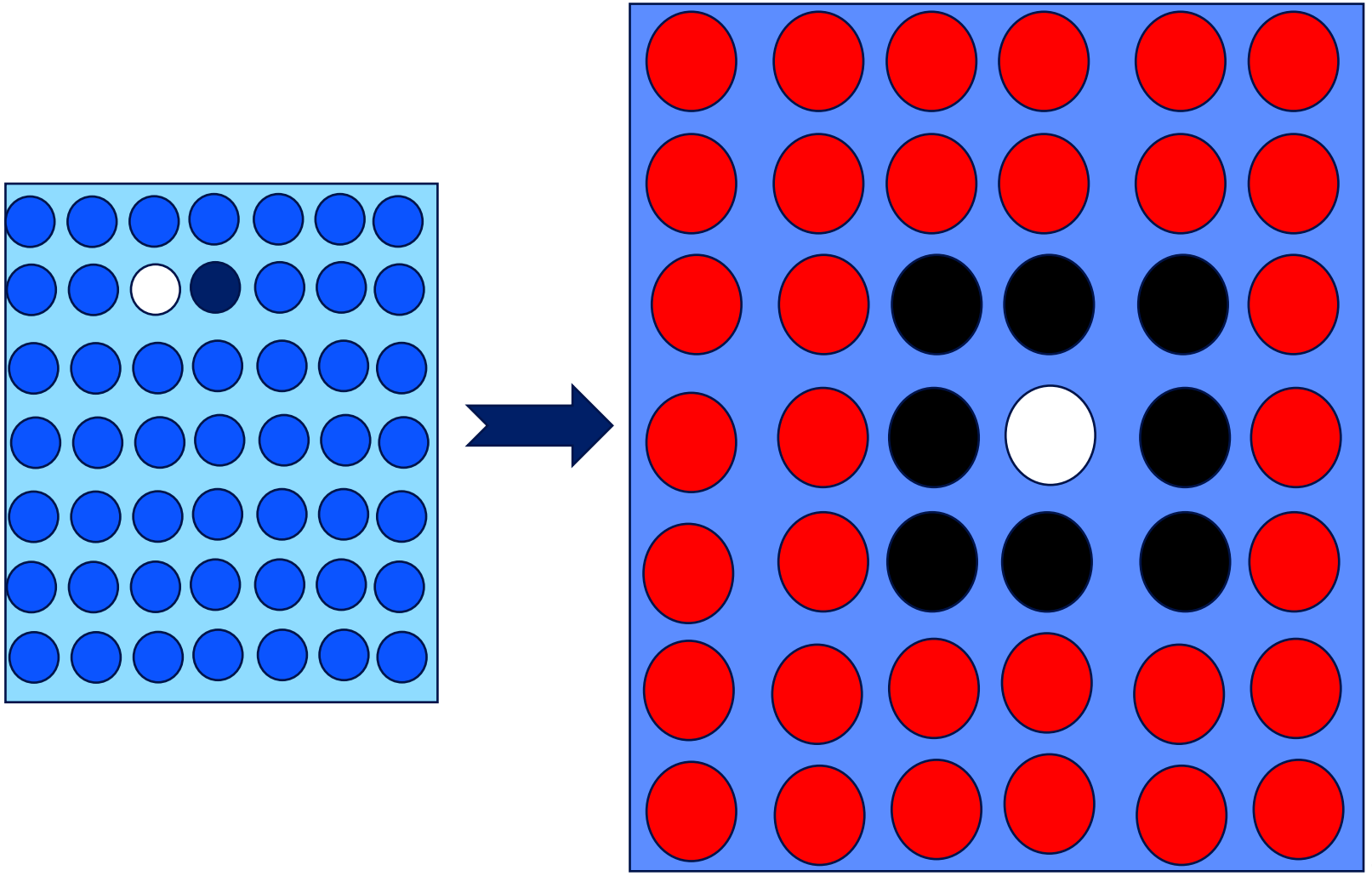
Q: sample unbalance ?

Demo #1

Is ML Universal ?

Limitations

- NFL
- Smooth
- Boundary



Workflow

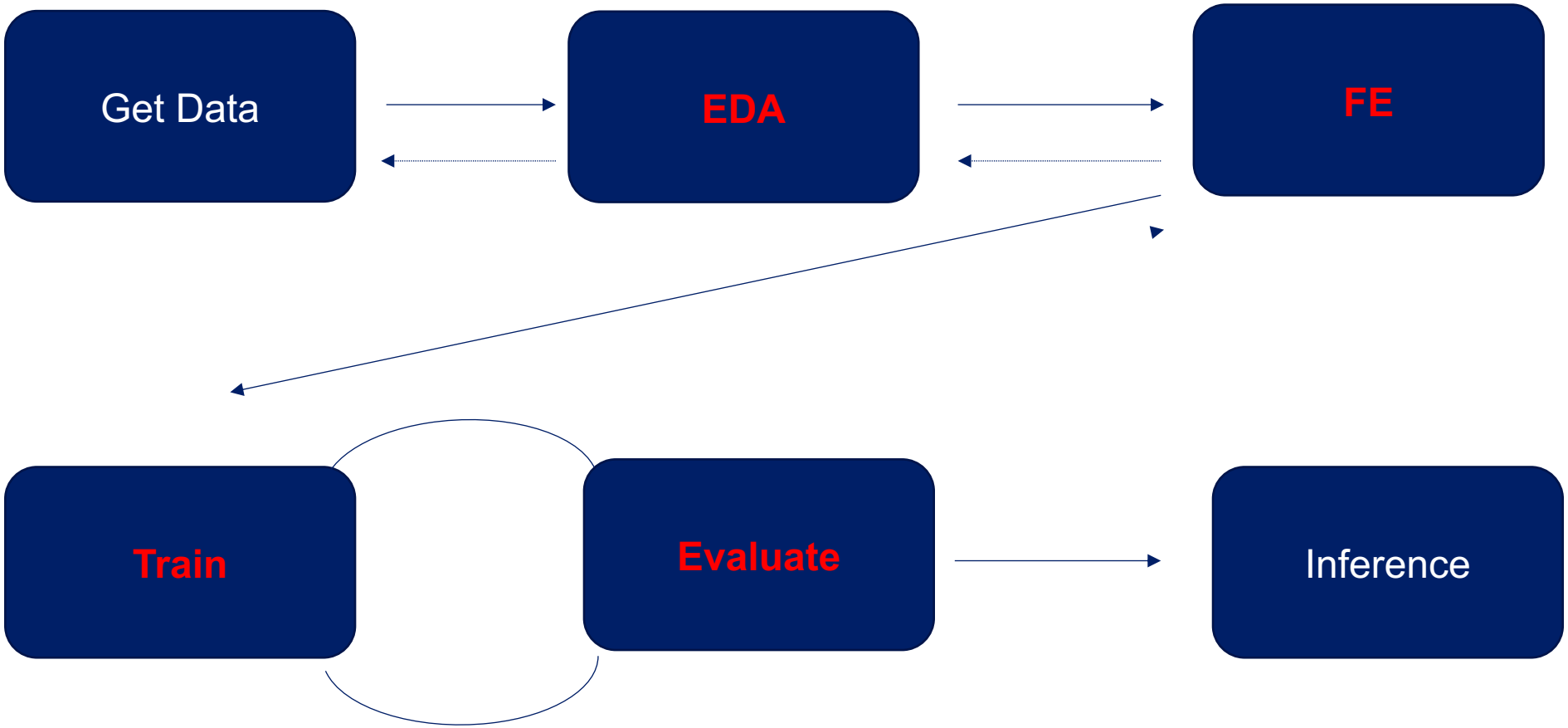
Get Data

Data cleaning

Train model

Use model

Workflow



Sample/Feature

Features

Samples

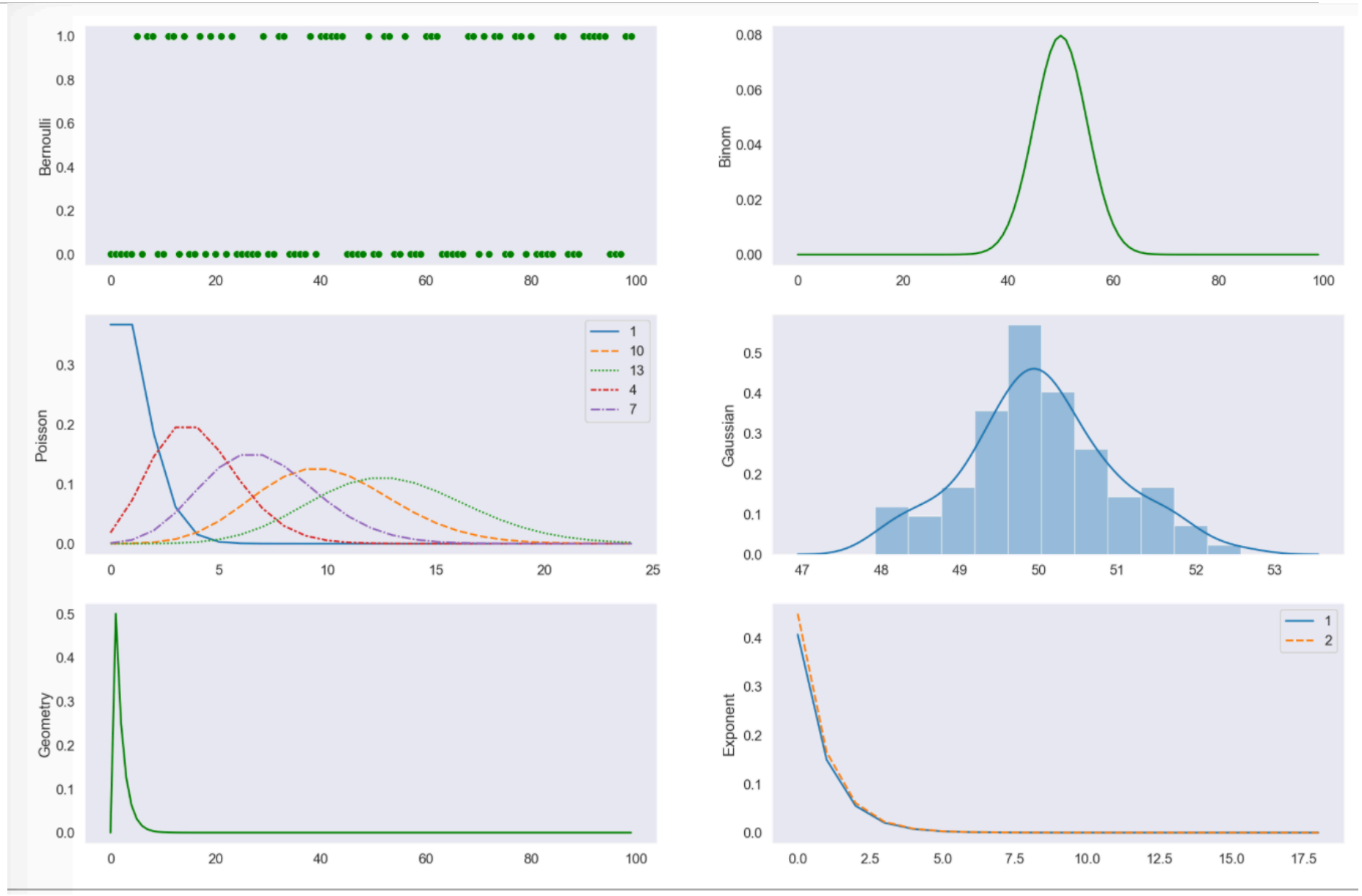
Log Time	User Name	Host	Action	Result
2017/1/6 8:46	user1	xxxx.xxxx.xxx.1	Login	Successful
2017/1/6 8:47	user2	xxxx.xxxx.xxx.2	Register	Failed
2017/1/6 8:47	user3	xxxx.xxxx.xxx.3	Set policy	Successful
2017/1/6 12:41	user4	xxxx.xxxx.xxx.4	Register	Successful
2017/1/11 10:28	user5	xxxx.xxxx.xxx.5	Login	Successful
2017/1/11 10:29	user6	xxxx.xxxx.xxx.6	Register	Failed
2017/1/11 10:29	user7	xxxx.xxxx.xxx.7	Set policy	Successful
2017/1/11 10:30	user8	xxxx.xxxx.xxx.8	Register	Failed
2017/1/11 10:30	user9	xxxx.xxxx.xxx.9	Login	Successful
2017/1/11 10:33	user10	xxxx.xxxx.xxx.10	Login out	Failed
2017/1/11 10:34	user11	xxxx.xxxx.xxx.11	Query log	Successful
……	……	…….	…..	…….

EDA & Preprocess

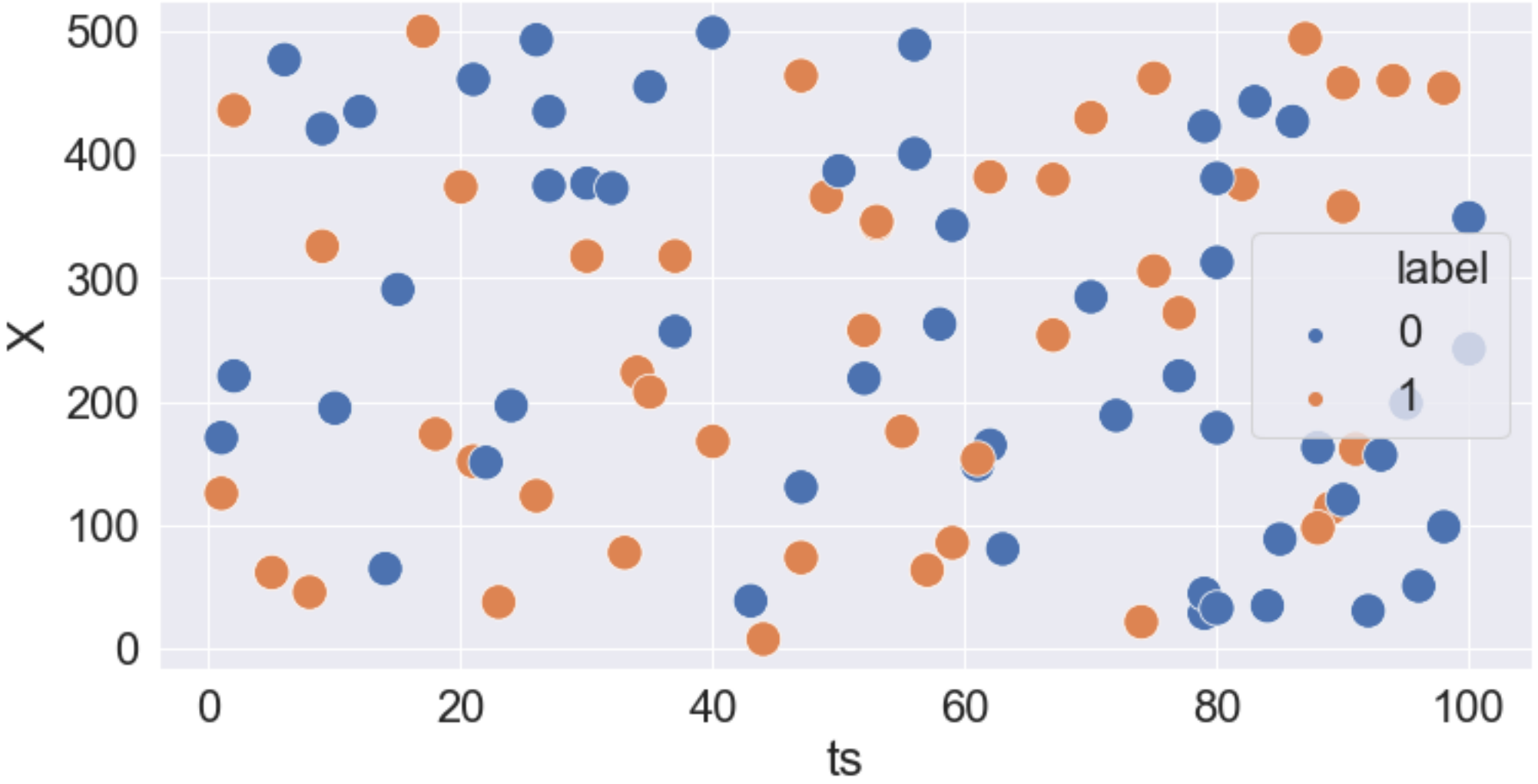
EDA #2

- Insight from graph
- Data distribution
- Normalization

Data distribution



Even-odd: distribution



Normalization

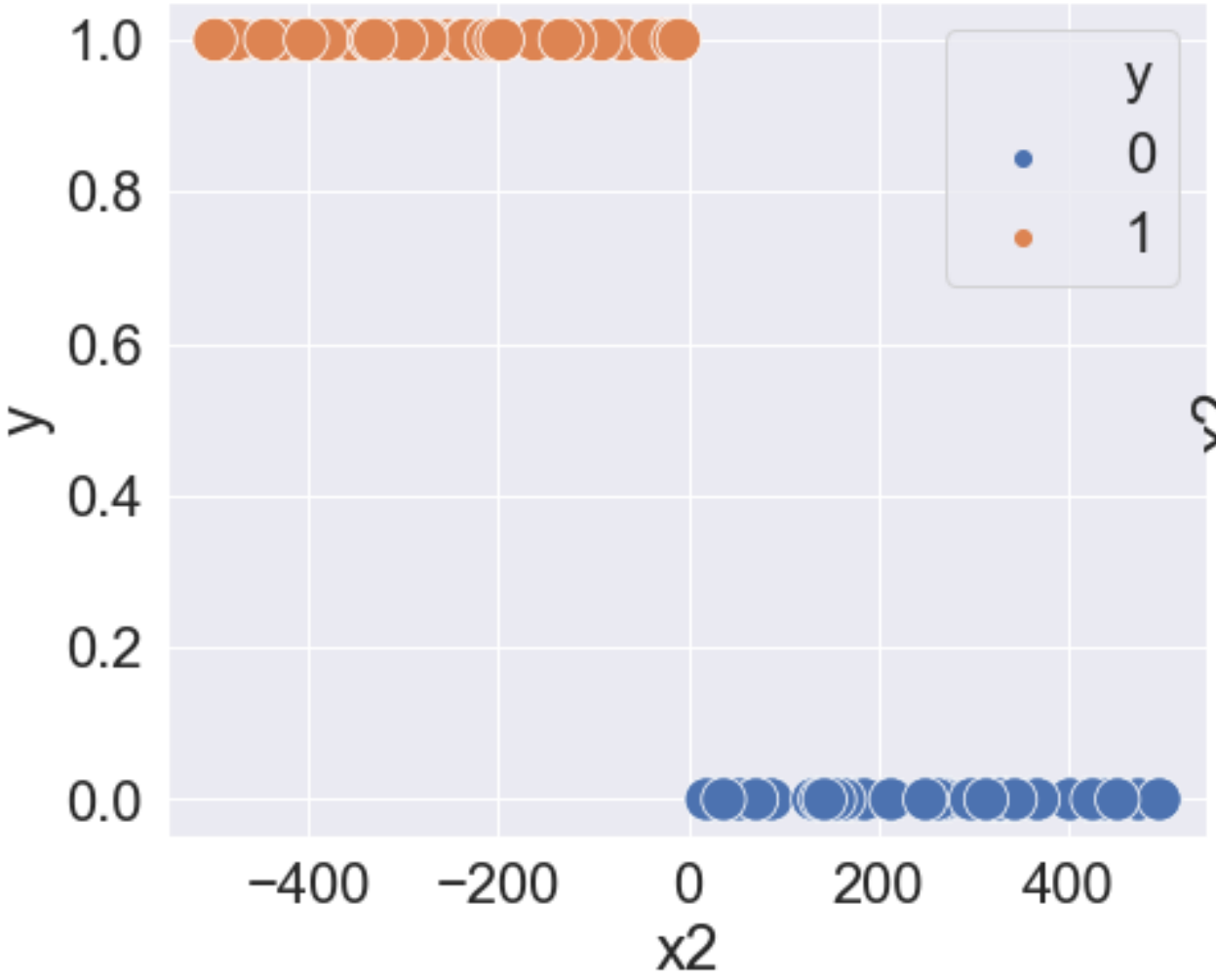
Why and How ?

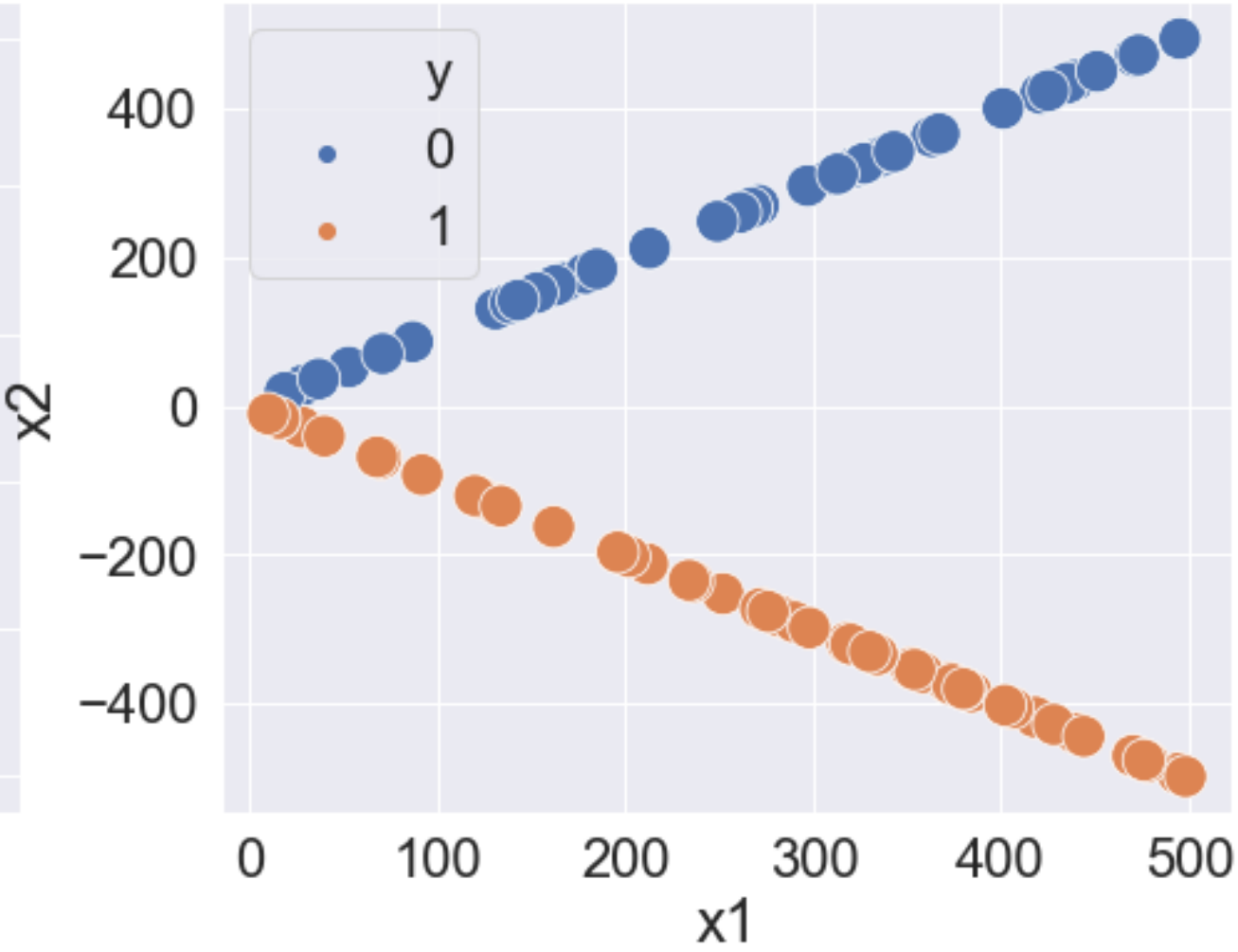
$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

$$x' = \frac{x - \mu}{\sigma}$$

Method	Data				
Raw data	[529	578	466	437	318]
MaxMinScale:	[0.81153846	1.	0.56923077	0.45769231	0.]
Z-score	[0.71550817	1.26850345	0.00451425	-0.32276867	-1.6657572]

Even-odd: New dimension by label



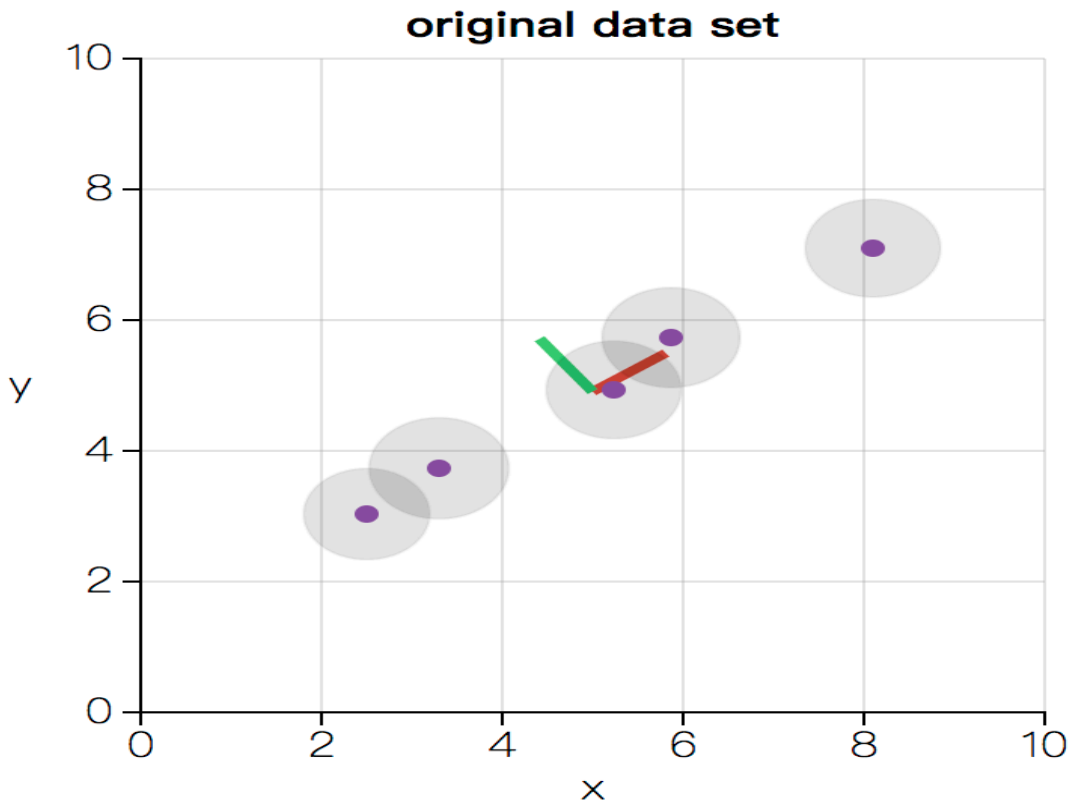


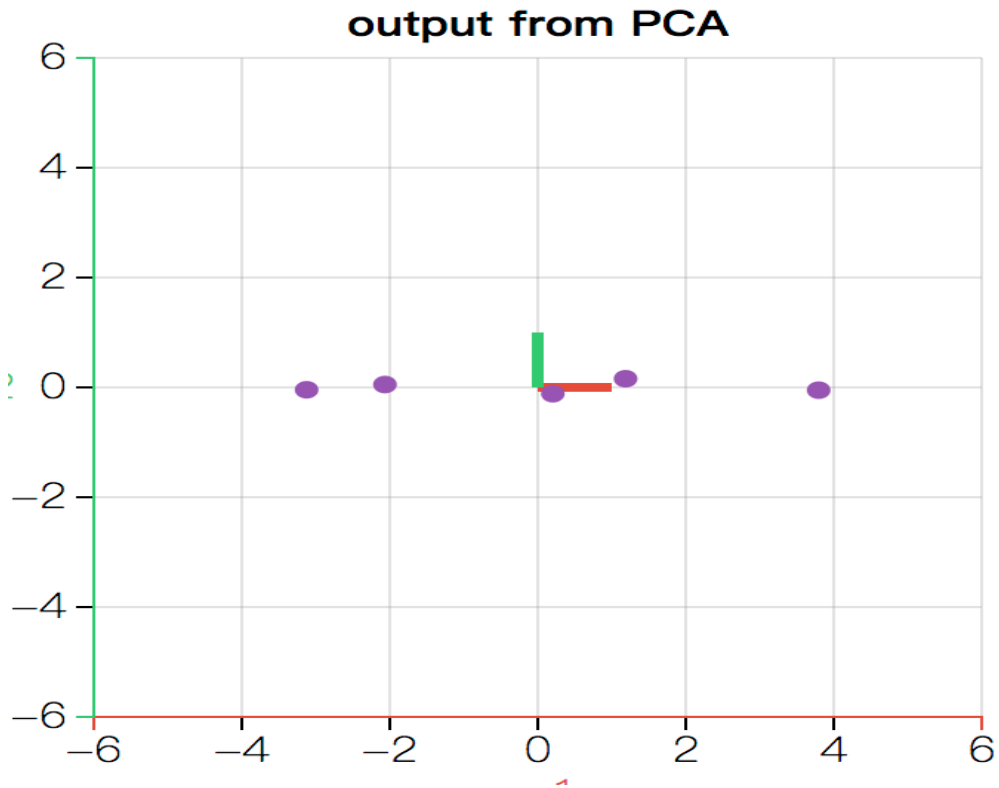
Feature Engine

FE #2

- Filter Methods
  - Correlation
  - ….
- Embed Methods
  - GA
  - ….
- Wrap methods
  - CNN
  - ….

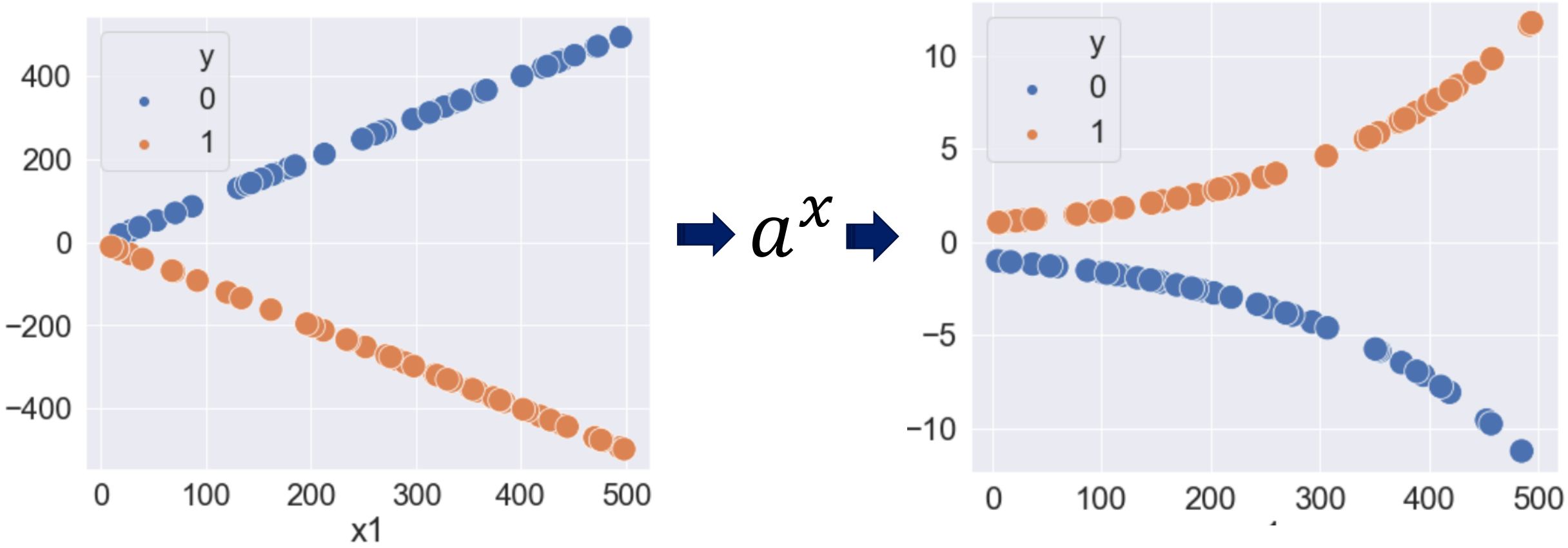
FE #3





https://setosa.io/ev/principal-component-analysis/

Even-odd: New dimension by transform



Train and Evaluation

Model and Evaluation

Models

- **Linear**
- DT/SVM/Bayes
- Boost/XGBoost
- **Neural network**
- **CNN**/RNN/LSTM
- ……..

Evaluate # Regression

Error = | Real – Predict |

$$RMSE(X,h) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(h(x_i) - y_i)^2}$$

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

$$MAE(X,h) = \frac{1}{m}\sum_{i=1}^{m}|h(x_i) - y_i|$$

Evaluate # Class

Confusion Matrix

	Predicted (Positive)	Predicted (Negative)
Actual (Positive)	TP	FN
Actual (Negative)	FP	TN

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

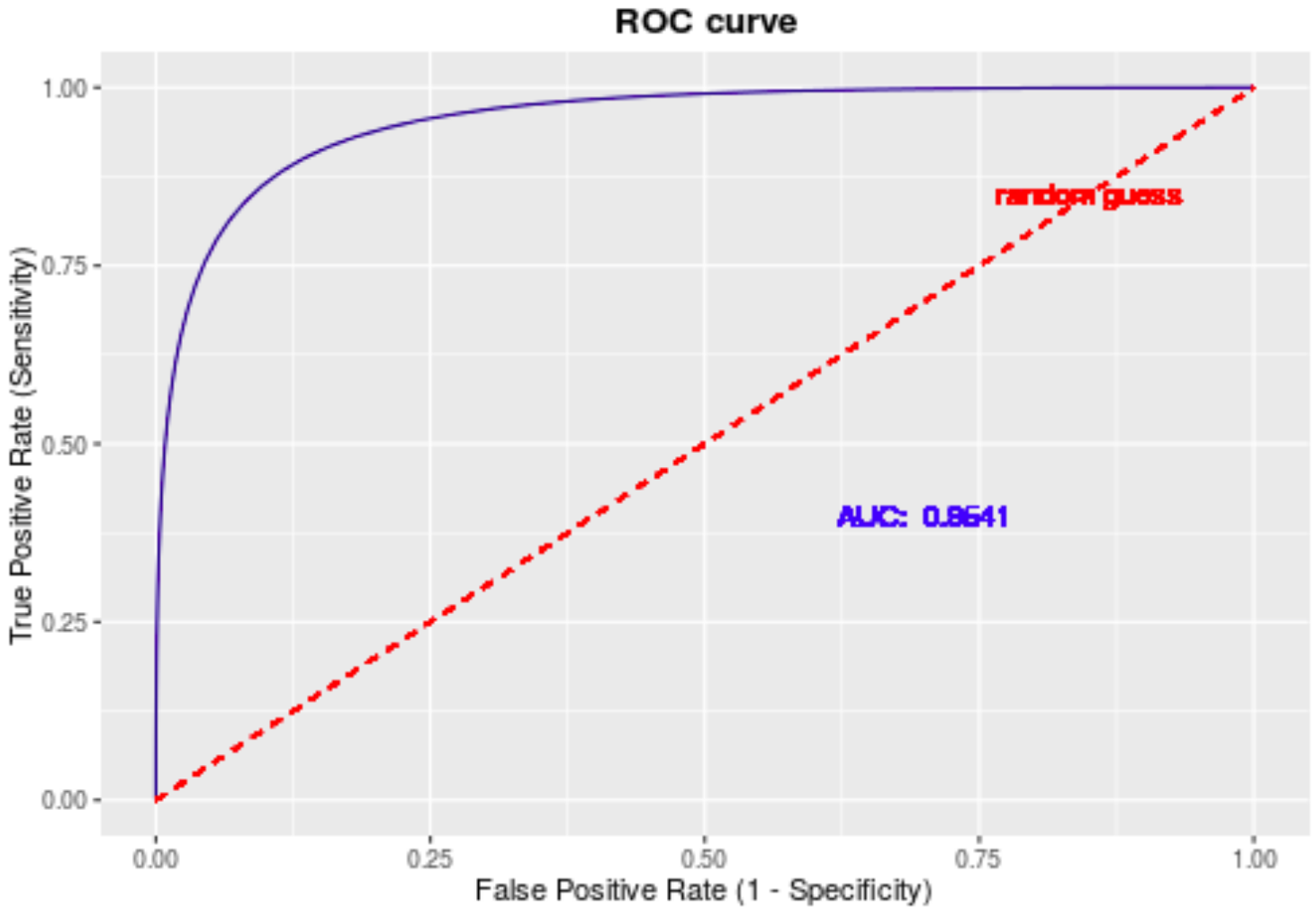
$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Sensitivity} = \text{Recall} = \frac{TP}{TP + FN}$$

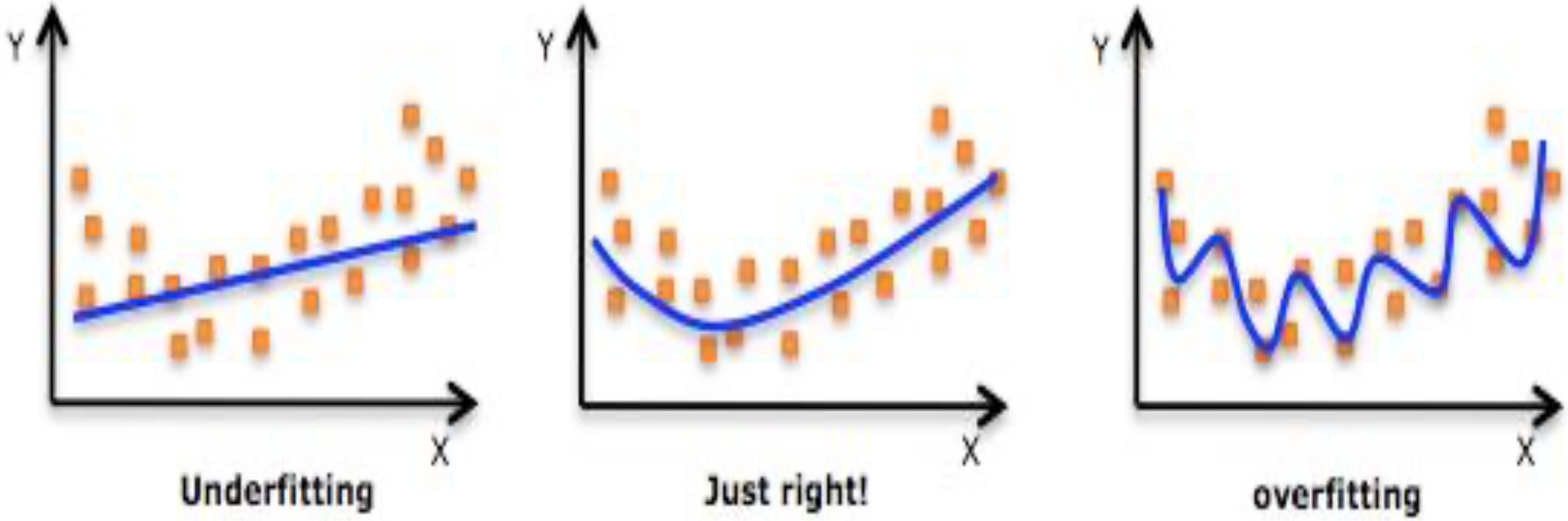
$$\text{Specificity} = \frac{TN}{TN + FP}$$

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precison + recall}$$

ROC


ROC curve
True Positive Rate (Sensitivity)
False Positive Rate (1 - Specificity)
0.00
0.25
0.50
0.75
1.00
random guess
AUC: 0.8641

Underfitting/Overfitting



Q: How to do ?

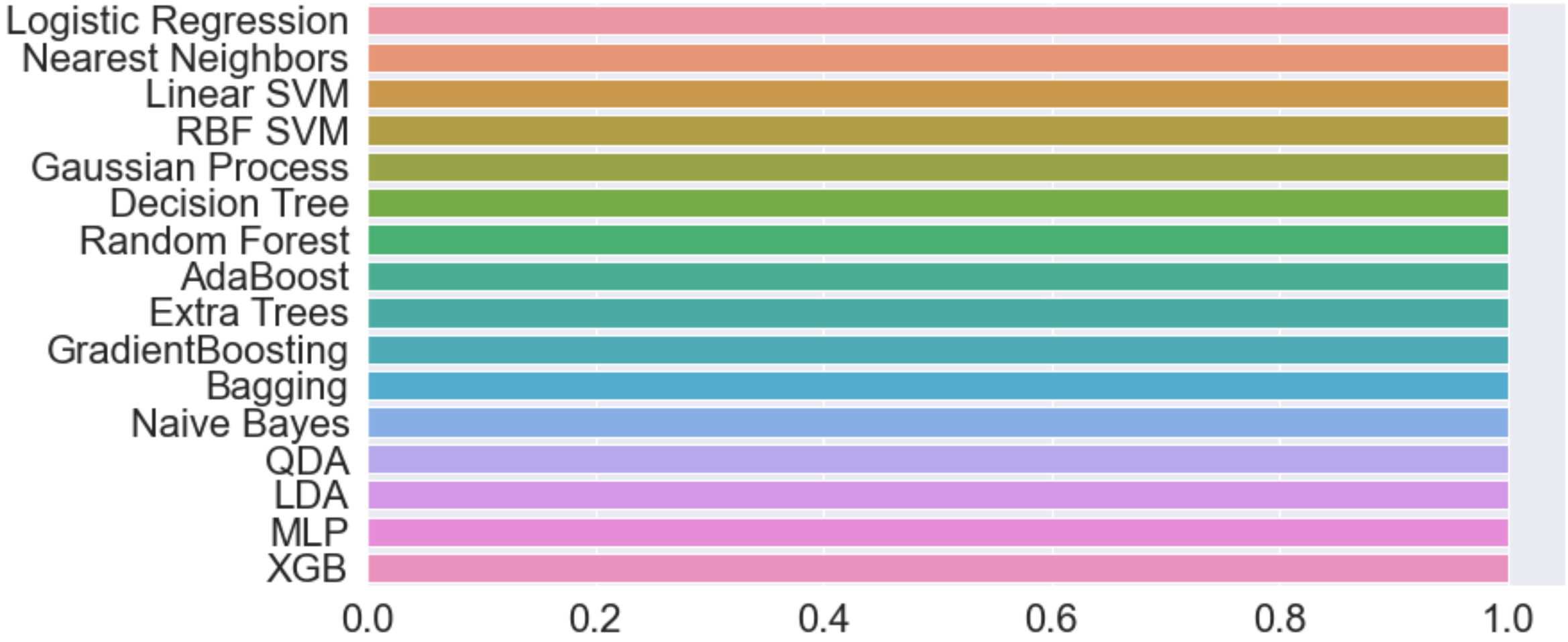
Even-odd: Train

```
train_x = [[ 2.05075082] [ 4.42065277] [ 1.13845955] [ 5.13414649] [ -
4.81182089] [ -3.56735022] [ 7.06474041] [ 1.07232113] [ -3.86369411]]
train_y = [[1] [1] [0] [1] [1] [1] [0] [1] [0]]

test_x =[[ 3.37690675] [ 4.46496981] [ -1.5432894 ] [ 1.893459 ] [
2.68460397] [ -2.64473427] [ 1.49033857] [ -2.80785569] [ 1.55100585]]
test_y = [[0] [0] [0] [0] [1] [1] [0] [1] [1] ]

from sklearn.tree import DecisionTreeClassifier
DecisionTreeClassifier(max_depth=5))
model.fit(train_x, train_y)
score = model.score(test_x, test_y)
```

Even-odd: Evaluate



Prediction

```
input_x = [[112],[244]]
input_x = np.random.random_integers(500,1000,20)
input_x=input_x[:,np.newaxis]
pred_x = np.array(input_x)
pred_x = prepress(pred_x)
pred_y = models['XGB'].predict(pred_x)
for x,y in zip(input_x,pred_y):
    if y == 1:
        print(" %d --> Even" %x[0])
    else:
        print(" %d --> Odd" %x[0])
```

```
896 --> Even
632 --> Even
721 --> Odd
932 --> Even
996 --> Even
985 --> Odd
725 --> Odd
881 --> Odd
838 --> Even
984 --> Even
906 --> Even
693 --> Odd
876 --> Even
867 --> Odd
503 --> Odd
995 --> Odd
923 --> Odd
580 --> Even
900 --> Even
808 --> Even
```

Brief summary

- ✓ More effort in EDA and FE
- ✓ Model has pre-condition and limitation
- ✓ Try several models or parameter pairs
- ✓ Evaluate and select suitable model

Thanks

2020-8-15

Backlog

Back log