Voting Loss function

Zekun zhao

MSE loss [mean square error]

1. take the **difference** between your predictions and the ground truth
2. **square** it
3. **average** it out across the whole dataset

Why it is not good enough?

Abnormal data affect normal data decision.

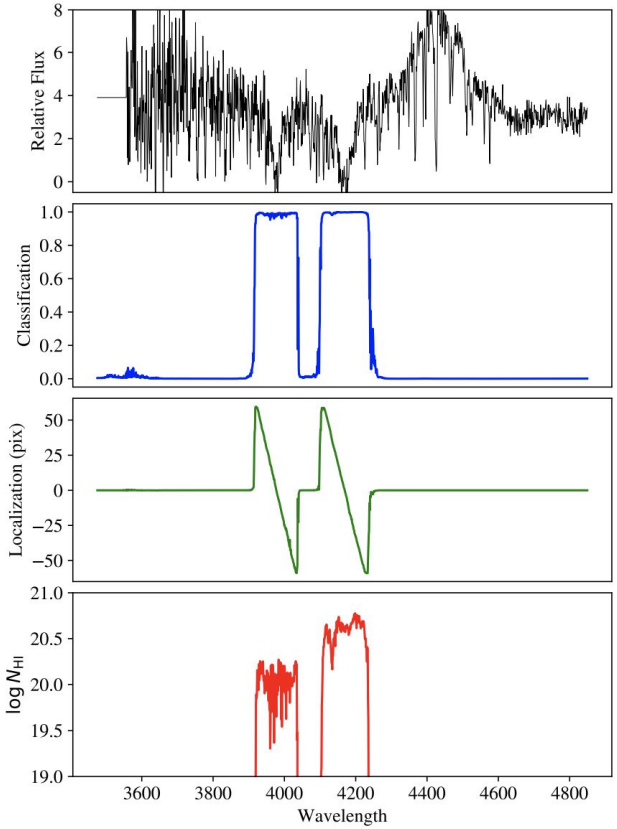
Not smooth; Bias Introduced.

Motivation

Rich set features:

1, Classification

2, Localization



Parks, D., Prochaska, J. X., Dong, S., & Cai, Z. (2017). Deep Learning of Quasar Spectra to Discover and Characterize Damped Lya Systems. *arXiv preprint arXiv:1709.04962*.

Motivation

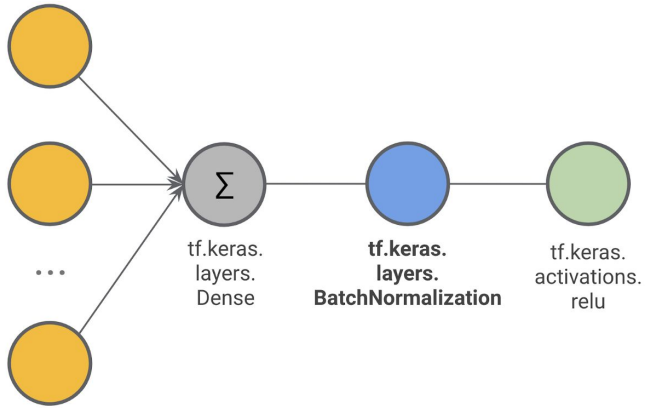
Regularization:

1, L2 :

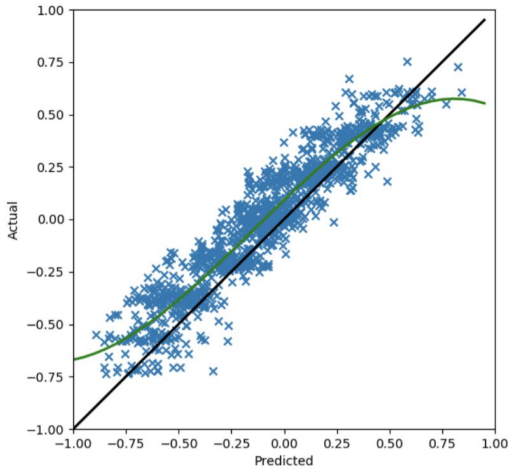
2, L1

3, Batch Normal **Regularization**

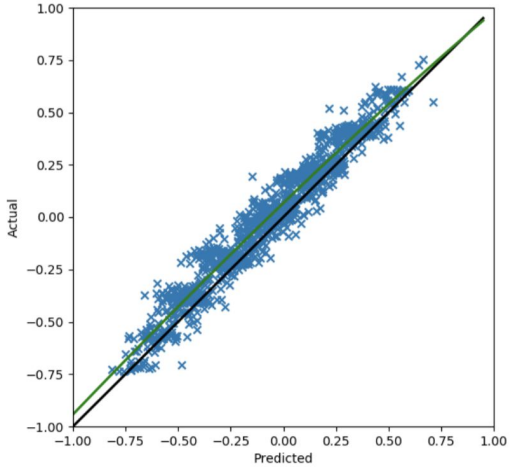
$$\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$



Motivation



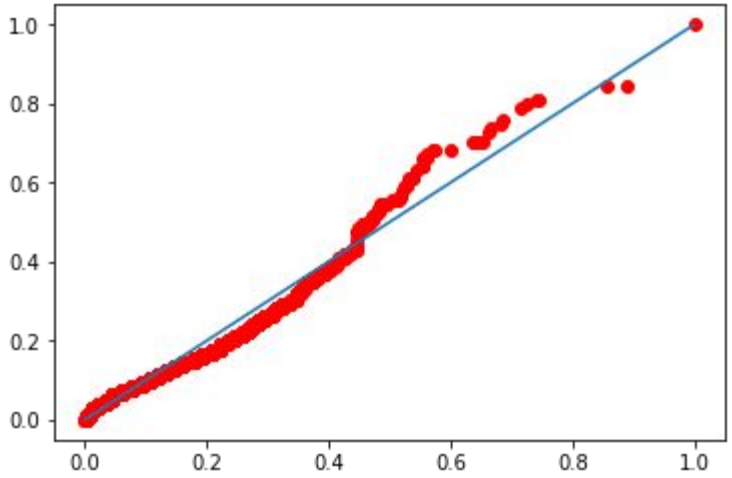
MSE



Voting loss

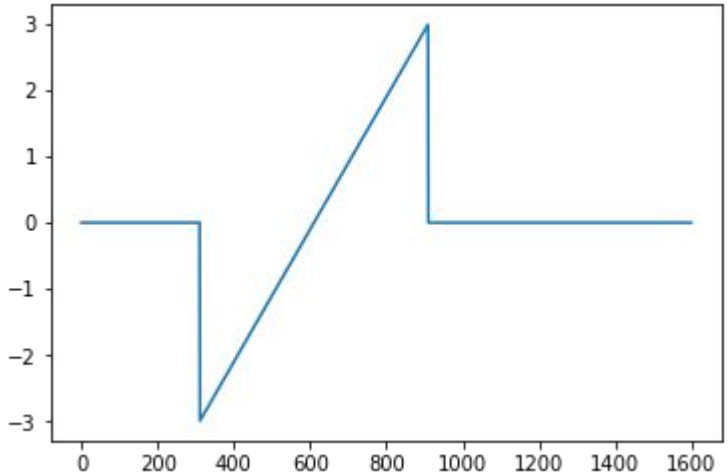
Experiment result from David Park’s current work.

True VS Preds



Basic Idea: Convert label to a high dimension

600 =>



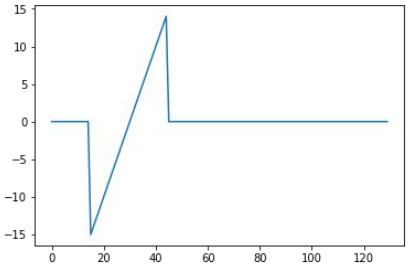
How voting works?

Giving one Apple to only one of N people;

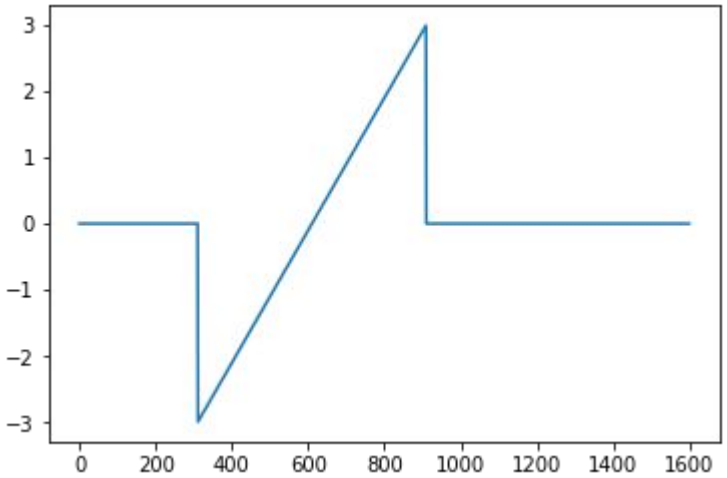
Each people to **give a vote** for who they think which one people has the Apple;

-1 votes for his/her **right 1 position** ; **+1** votes for **his/her left 1 position**

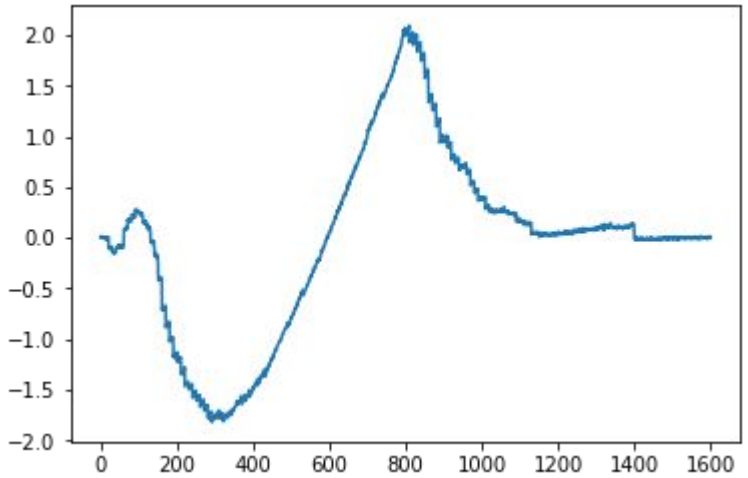
Voting_Value	0	-3	-2	-1	0	+1	+2	+3	0	0
Index_People	1	2	3	4	**5**	6	7	8	9	10



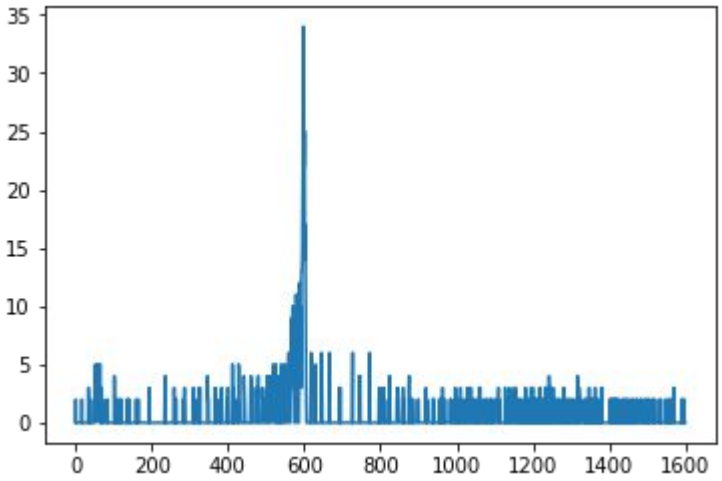
Label


3
2
1
0
−1
−2
−3
0
200
400
600
800
1000
1200
1400
1600

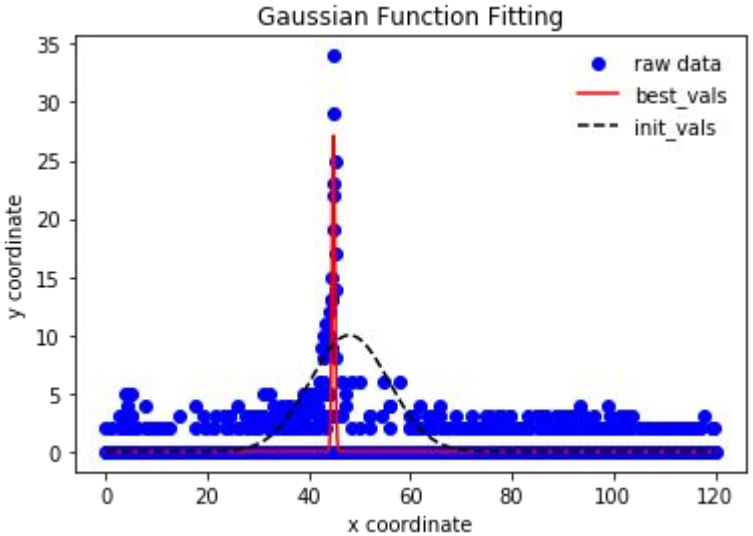
preds


2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
200
400
600
800
1000
1200
1400
1600

Voting Result

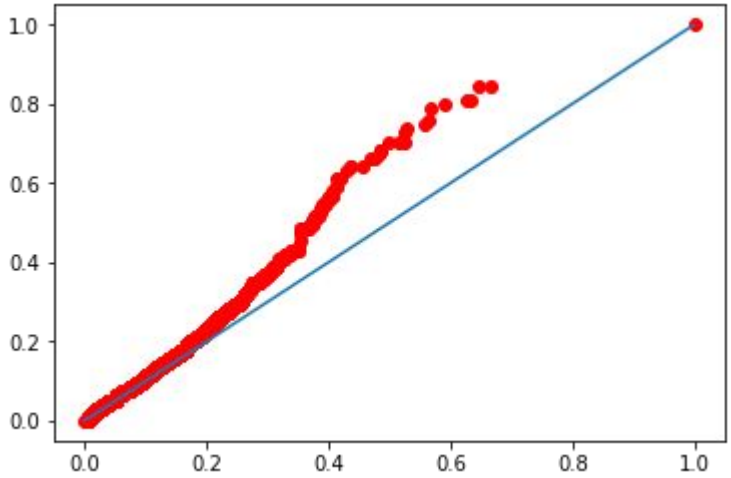


Fitting

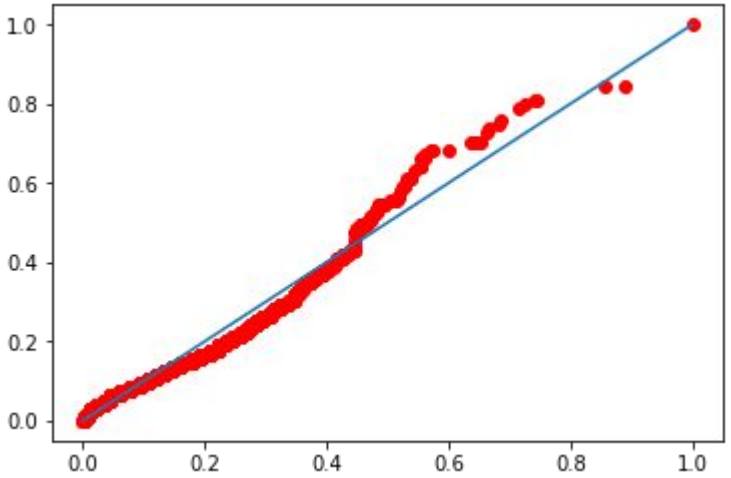


compare

MSE

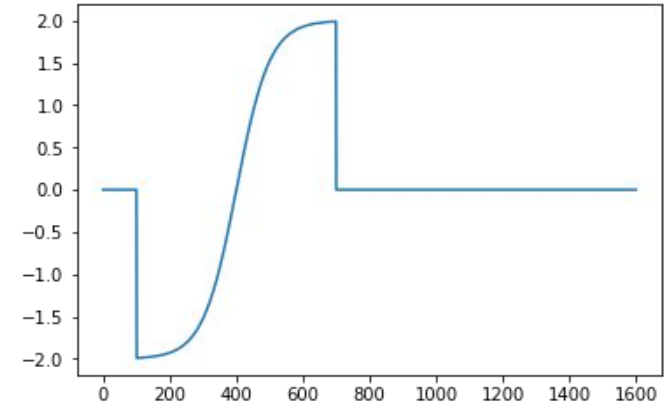


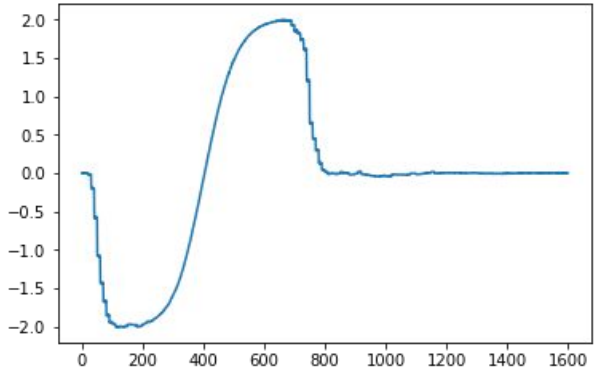
Voting loss

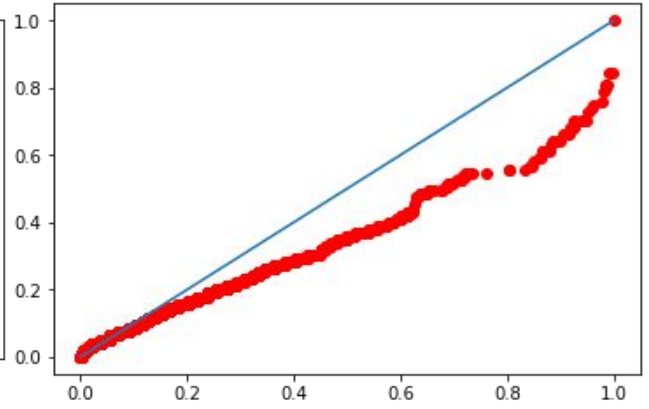


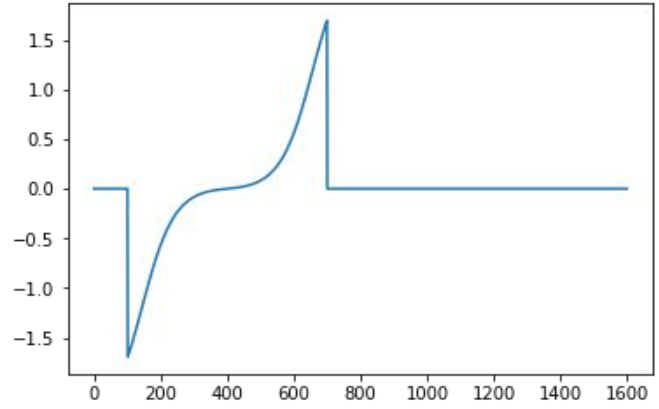
Techniques:

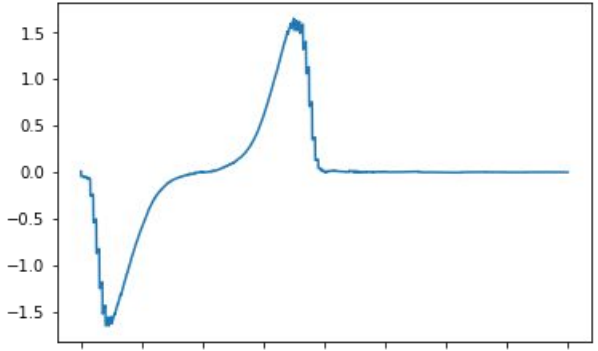
1. **Different Voting Strategy(linear or non-linear)**
2. Model Fitting Method (GMM)
3. Multi-label for one element(assign two Apple to two people or more Apple)
4. Adapted Parameters (mask, length)

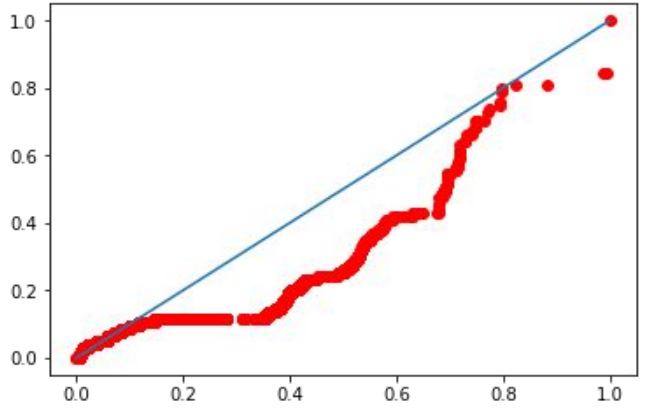

2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
200
400
600
800
1000
1200
1400
1600

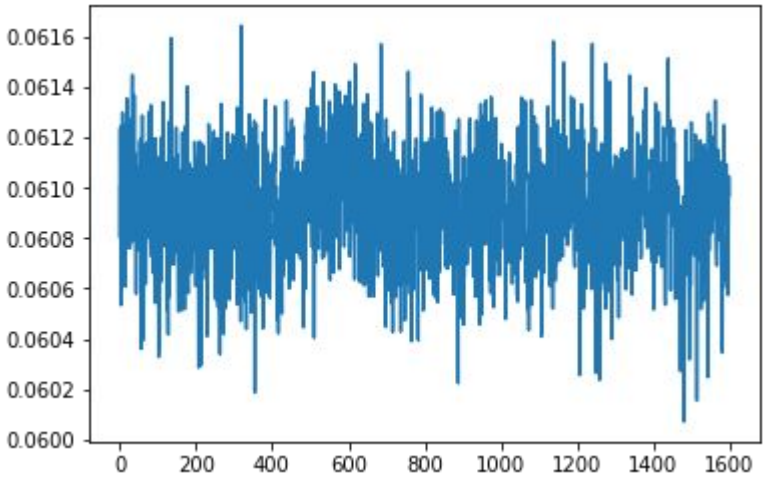

2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
200
400
600
800
1000
1200
1400
1600

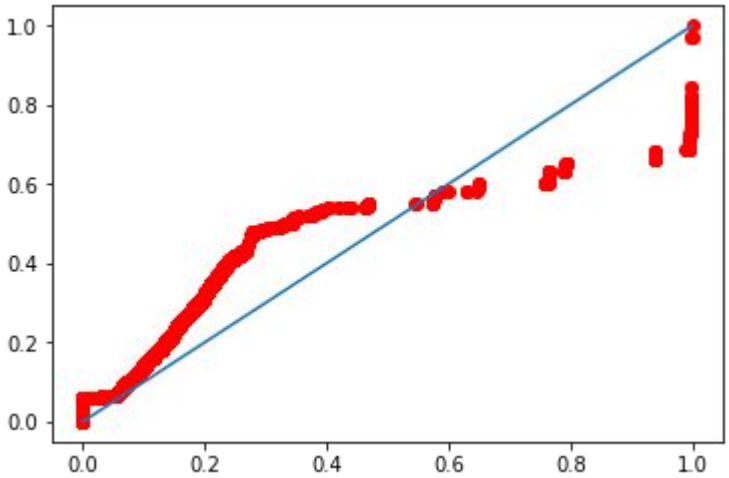

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
200
400
600
800
1000
1200
1400
1600


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
200
400
600
800
1000
1200
1400
1600


1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

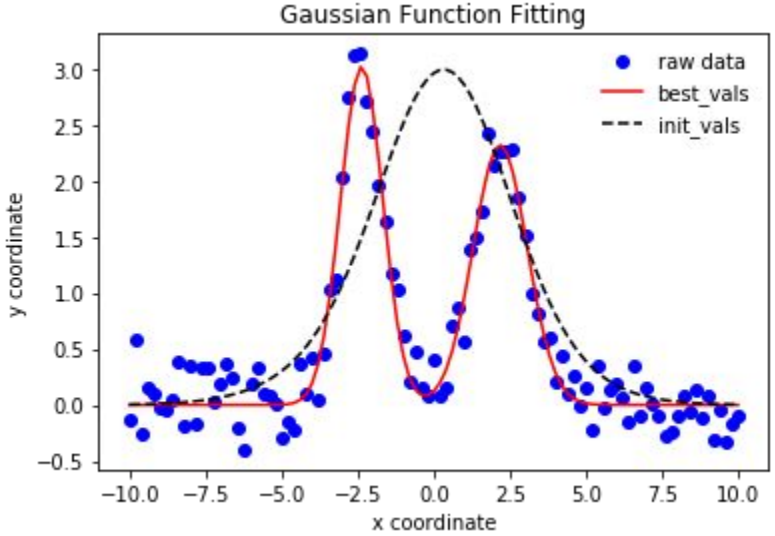

0.0616
0.0614
0.0612
0.0610
0.0608
0.0606
0.0604
0.0602
0.0600
0
200
400
600
800
1000
1200
1400
1600

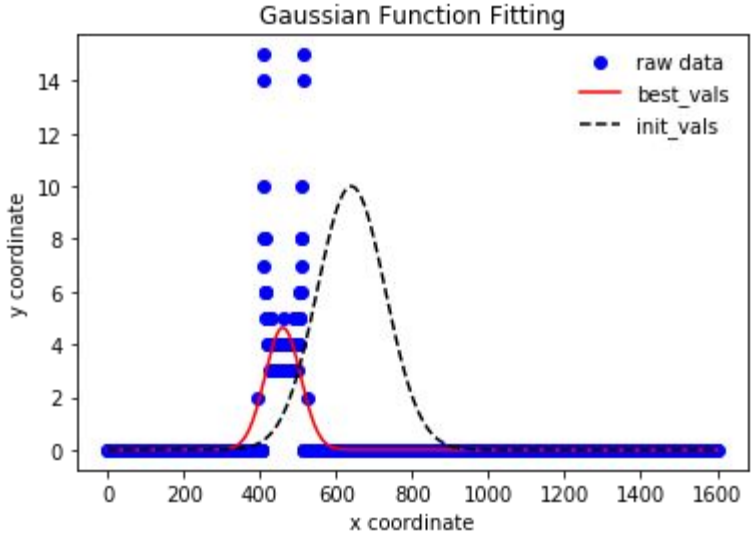

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

Techniques:

1. Different Voting Strategy(linear or non-linear)
2. **Model Fitting Method (GMM)**
3. **Multi-label for one element(assign two Apple to two people or more Apple)**
4. Adapted Parameters (mask, length)

Model Fitting Method (GMM)

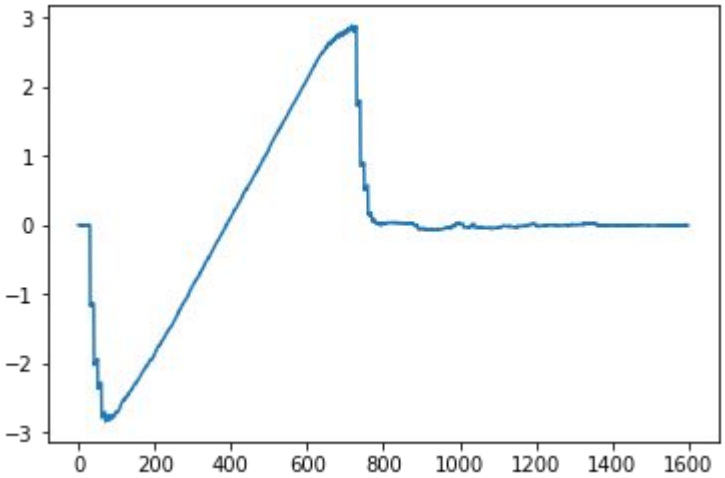


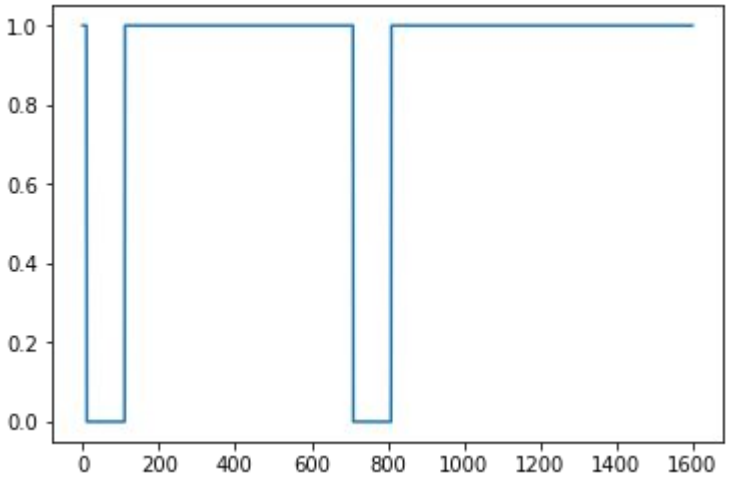


Techniques:

1. Different Voting Strategy(linear or non-linear)
2. Model Fitting Method (GMM)
3. Multi-label for one element(assign two Apple to two people or more Apple)
4. **Adapted Parameters (mask, length)**

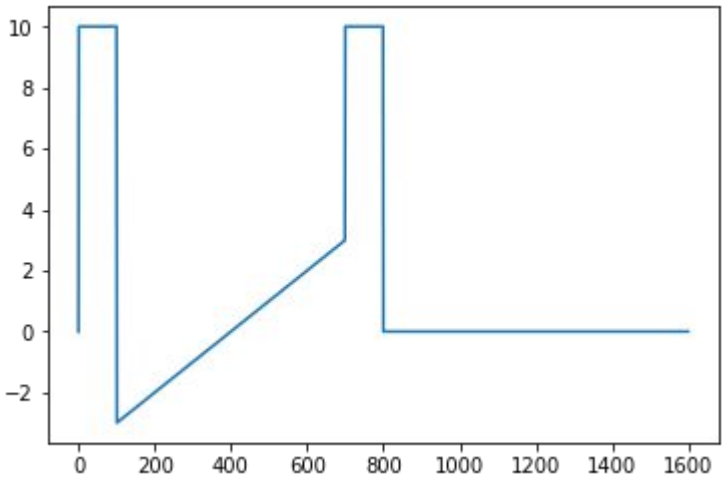
Adapted Parameters (mask, length)





$$L = \left(y' - y\right)^2 * \text{mask}$$

Speed up in loss calculation



tf.boolean_mask

y1
y2
y3
y4
y5
y6
y7
y8
y9
y10

→

y1
y2
y5
y6
y7
y10

→

$$MSE = \frac{\sum_{i=1}^{n} (y_i - y_i^p)^2}{n}$$

cons

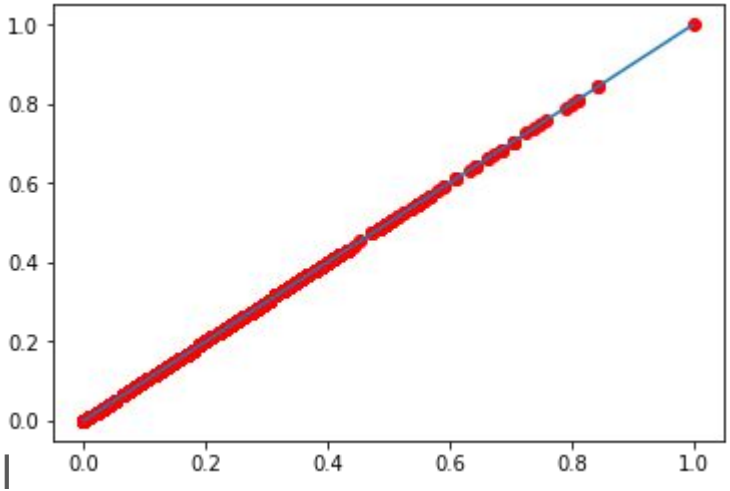
Distribution of dataset, abnormal data;

Difficult problem.

Special signal detection from sequence signal

multi-Object detection

saliency eye tracking



Thank you!