YIQING MA 2017/8/23

Using FNN model to predict CTR

1 EXPERIMENT SETUP

Data I evaluate the model based on the **All features.csv**. It is based on the corporation real data with each ad display information ad corresponding user click feedback.

- The all features dataset is **109 MB** data with **1970933** negative label (unclick) and **5047** positive label(click)in total.The features for each data instance are all categorical.Feature examples in the ad data are sn,dtTime,Province,City,District,Channel,Domain,Slot,Tag,Hour, Terminal,Agent,Material,fltCPM,boolClick.Among these, **10** features **City,District,Channel ,Domain,Slot,Tag,Hour,Terminal,Agent,Material** are chosen to predict Click.

- After one-hot encoding,the number of binary features is **1975980** in the whole dataset.We feed each compared model with these binary-feature data instances and user click(1) and non click(0) feedback as the ground-truth labels. In our experiments we using two dataset respectively.

- **small dataset 30K**

- **whole dataset All**

Models I compare the performance of the following two CTR estimation models.
FM: Factorisation Machine is a non-linear model able to estimate feature interactions even in problems with huge sparsity.
FNN: Factorisation-machine supported Neural Network is the Neural network initialized by a Factorisation Machine.
the experiment code of FNN is implemented with Theano.

Metric To measure the CTR estimation performance of each model, we employ the area under ROC curve(AUC) and the Root Mean Squared Error(RMSE).Both the **AUC** and **RMSE** and **Logloss**are widely used measure for evluating the CTR performance.

2 Performance Comparison

Table 2.1 shows the results compare FM and FNN on the small **30K** dataset and the total **190K** dataset.From the table,I observe that FNN is better than FM on all tested datasets.We can draw that based on the latent structure learned by FM,**FNN futher learns effective patterns between these latent features and provides a improvement over FM.**

Dataset	FM	FNN
AUC	2.29	2.08
RMSE	2.29	2.08
LogLoss	2.29	2.08

Table 2.1: CTR estimation on All Features data set performance.

3 Hyperparameter Tuning

Due to the fact that deep neural networks involve many implementation details and need to tune a fairly large number of hyper-parameters.Following is the details in tuning the hyper parameters.

Stochastic gradient descent is used to learn most of our parameters for all proposed models.Regarding selecting the number of training epochs, **Early stopping** is used. i try **different learning rate** from 1,0.1,0.01,0.001,to 0.001 and compare the performance and the convergence time consuming.The convergence time equals to the model training time for each epoch.

Table 3.1 shows that learning rate=0.01 is the best choice for the least convergence time.

serial	Learning Rate	Convergence Time	AUC performance
1	0.001	2.26	60
2	0.01	2.08	60
3	0.1	2.08	60
4	1	2.08	60

Table 3.1: Relationship between Learning Rate and convergence time.

4 Analysis of Parameters

Architecture In the model of FNN, the architecture with 3 ,4 and 5 hidden layers by fixing all layer sizes and find the architecture with 3 hidden layers is the best in terms of AUC performance. a diamond shape of neural networks is choosed with(200,300,100)

K Value I compare the K Value used on the FNN and FM. From Table 4.1 we can see the performance from K=3,5,10 on FM and FNN. When K=10 tend to be the most effective.

serial	K Value	AUC of FM	AUC of FNN
1	3	60	60
2	5	60	60
3	10	60	60

Table 4.1: Compare the K VALUE.

Activate Function I compare the activation function performance on the

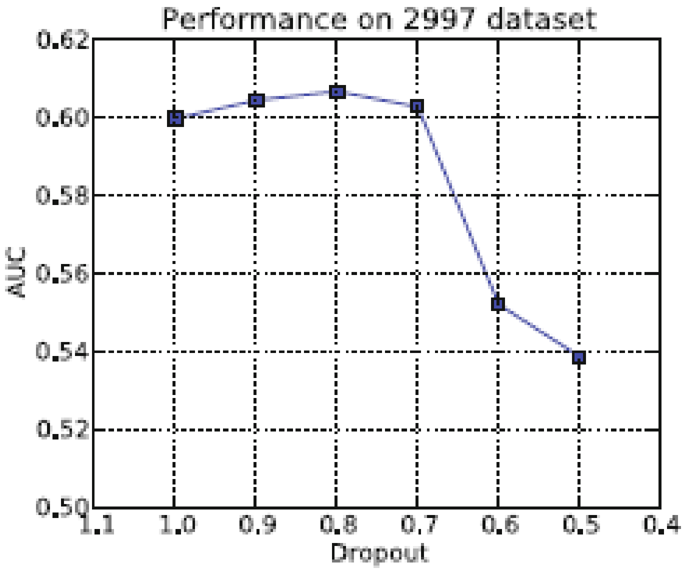
FNN. From Table 4.2 we can see that the sigmoid activation function tend to be the most effective

serial	Activate function	AUC performance
1	Sigmoid	60
2	Linear	60
3	Tanh	60

Table 4.2: Compare the Activate function.

DropoutDropout is also used to avoid overfitting.It is proved to have better performance than L2 regularisation

In order to show these two important parameters's relationship with the performance of the FNN model.Figure 1 show how the AUC performance changes with the increase of dropout rate. As being shown in the Figure, the dropout rate equals to 0.8 is the best.



(b) FNN on 2997 dataset

Figure 4.1: FNN on 30K dataset with different Dropout

5 CONCLUSION

Table 5.1: **Conclusion**

Best Parameters Setting	
Model	FNN
Data set	All Features
Learning Rate	rl=0.01
Layer	l=3
Structure	Dimond
Network	(200,300,100)
Dropout Rate	0.8
Activate Function	Sigmoid
Best Performance	
Time for Training per epoch	8S
Testing RMSE	0.6
Testing AUC	70