# 模型结果可视化分析工具

## 1 绘制PR曲线

为了能比较不同模型的优劣，这里利用绘制PR曲线来进行对比。

1.1 检测结果列表保存

**实现代码：PR\_statistic.py**

**主要参数：**  
model\_def = **'deploy.prototxt'** *# 检测网络*model\_weights = **'snapshot\_iter\_110000.caffemodel'** *# 训练好的模型*ROOTDIR = **"\\\\192.168.1.186/PedestrianData/"** *# 待测试样本集所在根目录*imgList = **"../Data\_0825/val.txt"** *# 样本列表*

**输出：**与模型同名的测试结果列表.txt文件

**测试结果列表格式：**

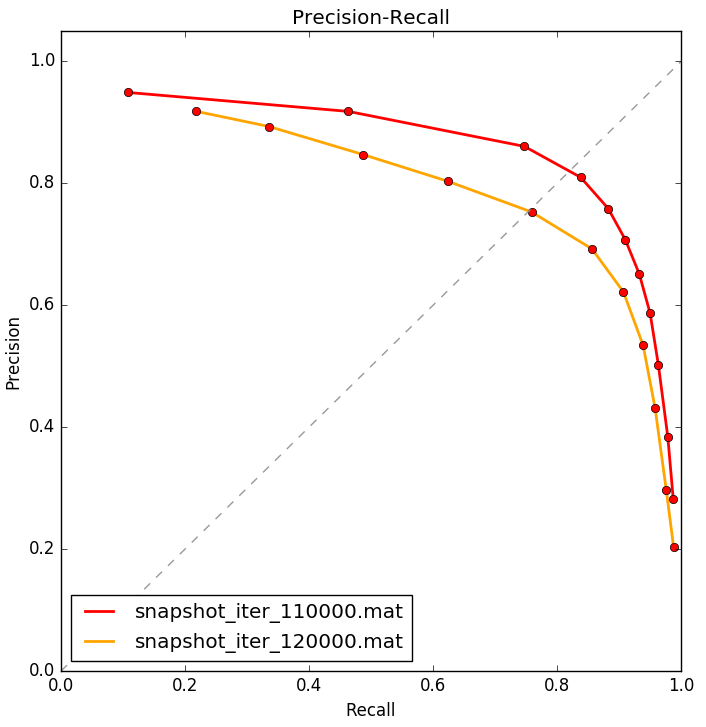
<**img\_name**> <<**prior\_boxes\_num**> <**confidence**> <**prior\_box1\_coordinates**>**...**>

1.2 统计并保存不同conf阈值下的TP、FP、FN，绘制PR曲线

**实现代码：PR\_statistic.py**

save\_data()函数读取上一步的检测结果，统计不同conf阈值下的TP、FP、FN值并保存成.mat格式文件

draw\_curve()函数读取mat文件绘制PR，曲线。第一个参数是要绘制的曲线数量（1-3），后面跟着不同模型下的统计结果。



## 2 绘制训练过程中loss、mAP曲线

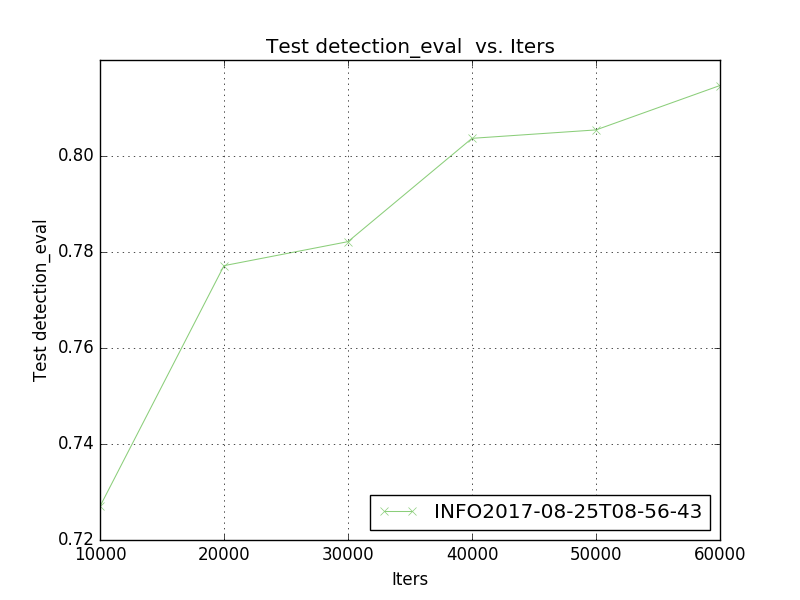
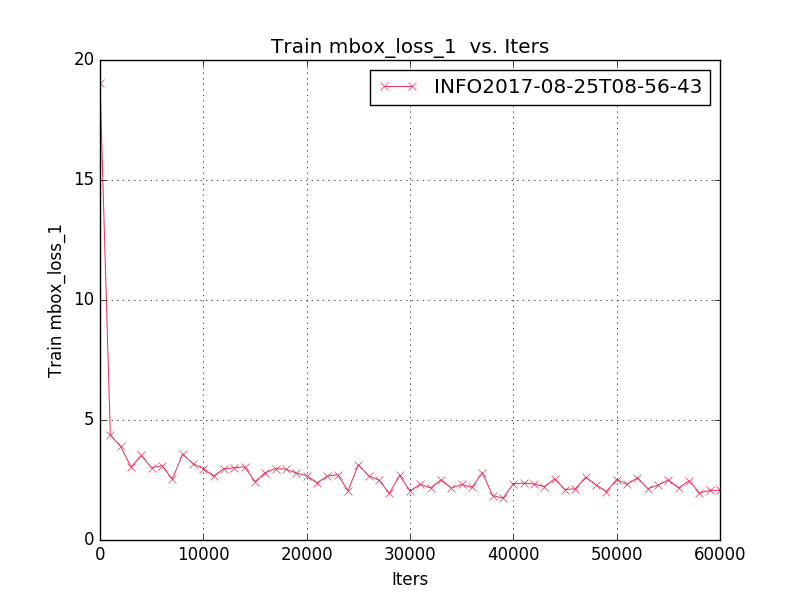
训练完成后，会在训练运行文件夹下的log文件夹下生成log文件，这个log文件存下的信息与训练时cmd窗口的输出信息一样。

**实现****代码：**plot\_training\_log.py、parse\_log.py、extract\_seconds.py

**主要参数：**

chart\_type = 0 *# 0：mAP, 4：learning rate, 6:loss*

path\_to\_logs = [**"INFO2017-08-26T14-00-20.log"**] *# log文件*



## 3 训练时输出test loss信息

1. 修改multibox\_loss\_layer.cpp中源文件在FindMatches()函数前加上

|  |
| --- |
| if (all\_match\_indices\_.size())  {  all\_match\_indices\_.clear();  all\_neg\_indices\_.clear();  } |

1. 修改solver.prototxt，添加

|  |
| --- |
| test\_compute\_loss: true |

1. 修改test.prototxt，在添加mbox\_priorbox\_1层之后添加

|  |
| --- |
| layer {  name: "mbox\_loss\_1"  type: "MultiBoxLoss"  bottom: "mbox\_loc\_1"  bottom: "mbox\_conf\_1"  bottom: "mbox\_priorbox\_1"  bottom: "label"  top: "mbox\_loss\_1"  include {  phase: TEST  }  propagate\_down: true  propagate\_down: true  propagate\_down: false  propagate\_down: false  loss\_param {  normalization: VALID  }  multibox\_loss\_param {  loc\_loss\_type: SMOOTH\_L1  conf\_loss\_type: SOFTMAX  loc\_weight: 1.0  num\_classes: 2  share\_location: true  match\_type: PER\_PREDICTION  overlap\_threshold: 0.5  use\_prior\_for\_matching: true  background\_label\_id: 0  #ignore\_label\_id: 13  use\_difficult\_gt: true  #do\_neg\_mining: true  neg\_pos\_ratio: 3.0  neg\_overlap: 0.5  code\_type: CENTER\_SIZE  mining\_type: MAX\_NEGATIVE  }  }  layer {  name: "silence"  type: "Silence"  bottom: "mbox\_loss\_1"  } |