Anomaly Detection

SRIHARSHA SURINENI

March 30, 2017

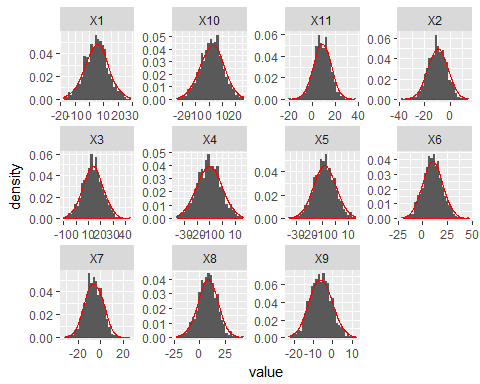
## Anomaly Detection:

The objective of this analysis is to implement an anomaly detection algorithm to detect anomalous behavior in server computers. The features measure the through- put (mb/s) and latency (ms) of response of each server. While servers were operating, m = 307 examples of how they were behaving was collected and thus have an unlabeled dataset fx(1); : : : ; x(m)g. You suspect that the vast majority of these examples are " (non-anomalous) examples of the servers operating normally, but there might also be some examples of servers acting anomalously within this dataset. You will use a Gaussian model to detect anomalous examples in your dataset. You will rst start on a 2D dataset that will allow you to visualize what the algorithm is doing. On that dataset you will t a Gaussian dis- tribution and then nd values that have very low probability and hence can be considered anomalies. After that, you will apply the anomaly detection algorithm to a larger dataset with many dimensions. You will be using ex8.m for this part of the exercise.

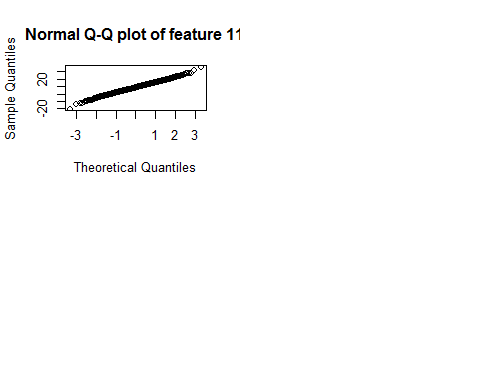
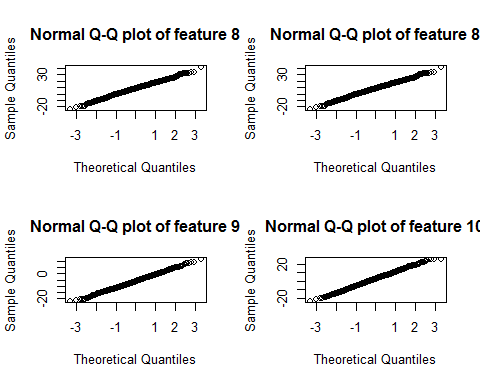
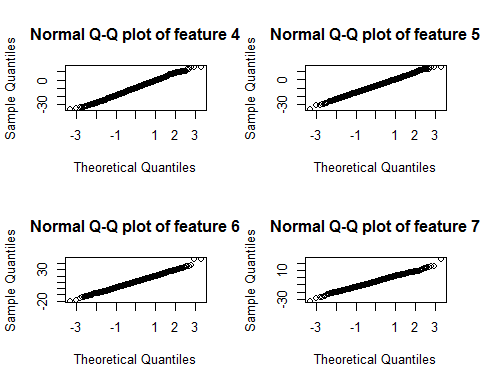
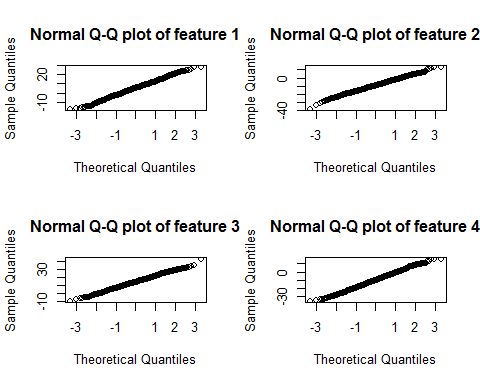
Exploratory analysis of the features:

## X1 X2 X3 X4   
## Min. :-17.4890 Min. :-38.594 Min. :-9.914 Min. :-35.968   
## 1st Qu.: -0.3169 1st Qu.:-14.749 1st Qu.: 8.574 1st Qu.:-16.857   
## Median : 5.3080 Median : -9.714 Median :13.684 Median :-10.443   
## Mean : 4.9394 Mean : -9.637 Mean :13.815 Mean :-10.464   
## 3rd Qu.: 9.9885 3rd Qu.: -4.862 3rd Qu.:18.648 3rd Qu.: -4.187   
## Max. : 28.3849 Max. : 13.690 Max. :43.322 Max. : 16.082   
## X5 X6 X7 X8   
## Min. :-35.352 Min. :-19.691 Min. :-32.2810 Min. :-23.218   
## 1st Qu.:-13.441 1st Qu.: 3.846 1st Qu.:-10.9270 1st Qu.: 1.903   
## Median : -8.032 Median : 10.126 Median : -6.1402 Median : 8.257   
## Mean : -7.956 Mean : 10.200 Mean : -6.0194 Mean : 7.970   
## 3rd Qu.: -2.511 3rd Qu.: 16.528 3rd Qu.: -0.3967 3rd Qu.: 14.181   
## Max. : 15.592 Max. : 47.132 Max. : 25.4738 Max. : 41.978   
## X9 X10 X11   
## Min. :-22.040 Min. :-23.459 Min. :-20.152   
## 1st Qu.:-10.089 1st Qu.: -3.480 1st Qu.: 3.954   
## Median : -6.278 Median : 2.550 Median : 8.479   
## Mean : -6.253 Mean : 2.325 Mean : 8.474   
## 3rd Qu.: -2.540 3rd Qu.: 7.780 3rd Qu.: 13.245   
## Max. : 12.090 Max. : 26.166 Max. : 36.545

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

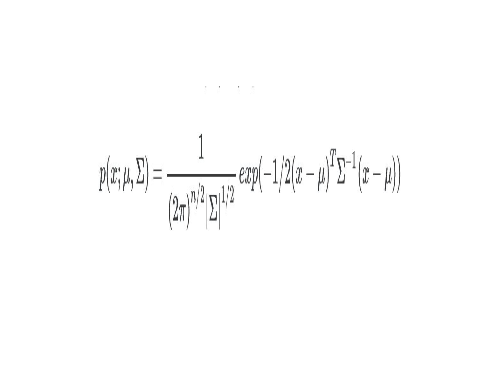


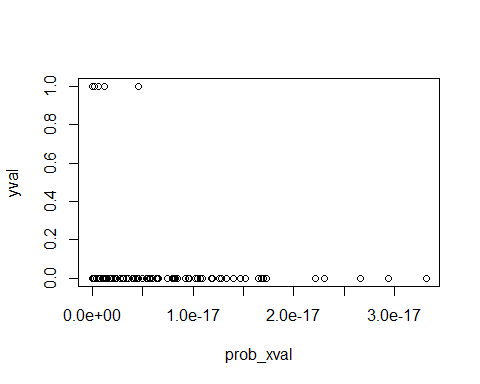
All the eleven features of the dataset look like they follow normal distribution from the above plots. Let's have a look at their QQPlots and resuts of tests of normality:

 p-values of the Shapiro and Anderson\_darling tests are as follows:

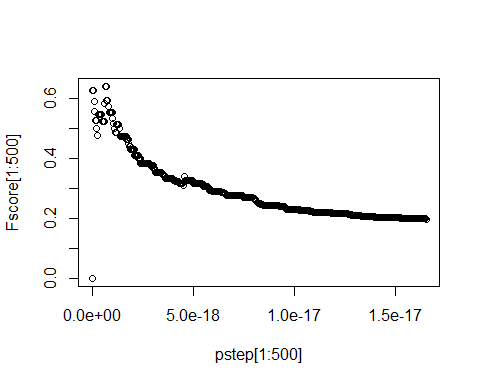
## X1 X2 X3 X4 X5 X6  
## shapiro 0.1609494 0.6697672 0.6582598 0.5722965 0.9618786 0.4980186  
## adtest 0.1207252 0.8379918 0.7837818 0.9635222 0.9756475 0.9267057  
## X7 X8 X9 X10 X11  
## shapiro 0.1774801 0.3881551 0.8004087 0.6326609 0.4665709  
## adtest 0.1469380 0.2843801 0.7784174 0.5199585 0.6737168

There is no evidence from these tests as well to suggest any deviance from our assumption of normality of the features. So, our basic assumption that the features are derived from normal distribution is valid. Anomaly detection algorithm models the joint probability distributon function as multivariate normal distribution:





## Choosing threshold:



From the above analysis, the threshold of joint probabiltiy dstribution function value to categorise an anomaly is 6.624241210^{-19} Misclassifcatin table for this threshold for validation set: 83, 7, 2, 8 F1 - score for this threshold: 0.64