Final-Project 2 - MATH6380J
LASSO Regression on Drug Sensitivity Data sets and the Classification of Art Paintings

DO Van Thuat (SID: 20405634), LO Yi-Su (SID: 20399988)
Department of Mathematics, School of Science
The Hong Kong University of Science and Technology
tdovan@connect.ust.hk

Abstract

In this report, we deal with various lasso regression methods for drug sensitivity prediction. In particular, we take distribution of output (i.e. cancer cell lines' response to drugs) into account. Using lasso regression, we experimented several distributions (inclusive: normal, poisson, gamma, inverse gaussian) and found that a more appropriate distribution makes a better fitting result. Some observations on PCA to reduce dimension are also provided. In addition, we make some effort for the Raphael's painting identification. A adequate result has not been available, however, the description and a potential method to this problem are presented in the second section for further discussion and improvement.

Contribution 50-50. *Each person contribute a half of the work, not only in experiments but also in report writing.*

1 Lasso regression on drug sensitivity datasets

In this section, we work with three datasets of drug sensitivity of cancer cell lines:

- 1-drug: responses of 542 cancer cell lines with 60 binary features (genes) to one drug, http://inclass.kaggle.com/c/drugsensitivity-2;
- 20-drugs: responses of 100 cancer cell lines with 20 discrete features (drug dosages) to a combination of 20 drugs, http://inclass.kaggle.com/c/combodrug20;
- Cleave-drug: responses of 129 cancer cell lines with over 18875 genetic features to drug in time-point: 24-48-72 hours, http://inclass.kaggle.com/c/drugsensitivity.

Based on learning from given samples, we target to predicts the responses (IC50 outputs) of the test samples. The difficulties when dealing with the datasets of drug sensitivity are:

- Inputs is discrete while outputs are continuous;
- It is really hard to predict continuous outputs with ambiguous distribution.

We observe that the output range of a dataset is normally a bounded interval on the real line. Distribution of sample-outputs (density) is abnormal. It is very concentrated on a certain subset of the range, but is also sparse on other intervals. It is natural to think that, if we are able to understand the underlying distributive properties of the outputs, then we can expect a good regression. Our limited consideration in practice is inclusive *normal, poisson, gamma and inverse gaussian distributions*.

Simply, we only use *lasso (and lasso generalized) regression* to fit the datasets. Our concern is mostly the influence of choosing various distribution on fitting results. For every experiment, we use *10-folds*

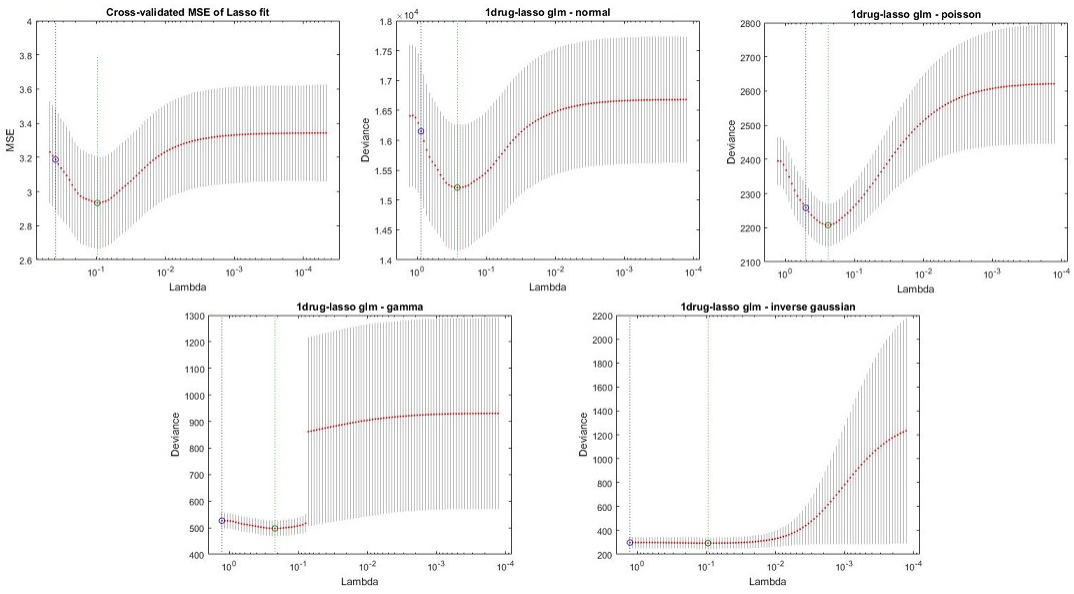

Figure 1: 1 drug dataset

cross validation to avoid over fitting, then choose a lambda parameter with minimum error. Note that *mean squared error (MSE)* in lasso fitting is equivalent to *deviance* in lasso generalized regression with respect to normal distribution.

1.1 On 1-drug samples

In this dataset, the original responses are not positive values at all, so we raise them to exponentiation of 2, i.e. a $new = 2^{old}$. This is a simple adaption to the requirement of distributions other than normal one. We see that gamma and inverse gaussian distributions improve the results of generalized lasso fits remarkably. The best one is generalized lasso associated with inverse gaussian distribution (that is the best fit to the response output). The following table shows the error of lasso fit (MSE) versus generalized lasso with respect to various distributions.

Table 1: one drug

lasso	normal	poisson	gamma	inverse gaussian
2.93	15207	2207	497	293

1.2 On 20-drugs samples

On this sample, we apply dummy representation for a drug with

- dosage level 0 is equivalent a triple (0, 0, 0),
- dosage level 1 is equivalent a triple (1, 0, 0),
- dosage level 2 is equivalent a triple (0, 1, 0),
- dosage level 3 is equivalent a triple (0, 0, 1).

This aims to transform the dataset into binary featured. However, The transformation makes the fitting worse (possessing higher MSE). Since the original responses are not positive values at all, we raise them to exponentiation of 2, i.e. a $new = 2^{old}$ to adapt the requirement of not-normal distributions.

We see that distributions other than normal one improve the results of generalized lasso fits a bit. The best one is generalized lasso associated with poisson distribution (that is the best fit to the response output). The following table shows the error of lasso fit (MSE) versus generalized lasso (deviance) with respect to various distributions.

lasso	lasso dummy	normal	poisson	gamma	inverse gaussian
0.00002	0.0168	0.0062	0.0012	0.0056	0.02

Table 2: 20 drugs

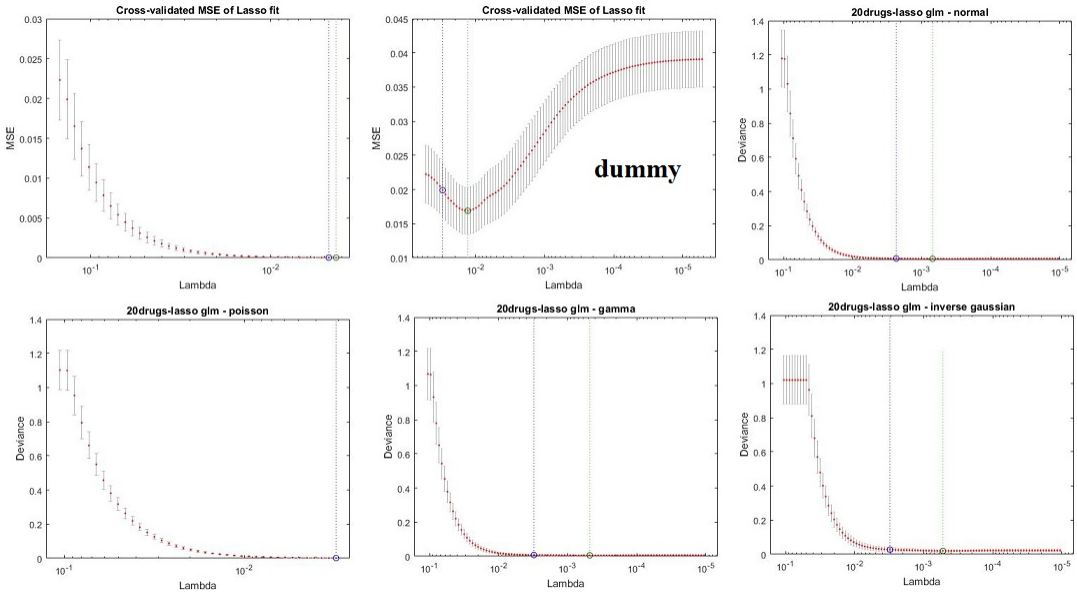

Figure 2: 20 drugs dataset

1.3 On Cleave-drug samples

In this dataset, we will apply PCA to reduce dimension. Then the timing costs less but accuracy doesn't improve. The following tables shows the error of lasso fit (MSE) versus generalized lasso (deviance) with respect to various distributions. PCA is also applied to observe the impact of dimensional reduction.

Table 3: 24 hours

pca	lasso	normal	poisson
no PCA	1100	139567	4805
PCA	1045	135863	4705

Table 4: 48 hours

PCA	lasso	normal	poisson
NO	420	53287	3140
YES	409	53409	3253

In Table 3 and Table 4, poisson distribution is much better than normal one (In deed, poisson distribution fits the response outputs best).

Table 5: 72 hours

PCA	lasso	normal	poisson	gamma	inverse gaussian
NO	0.95	124	66	52	nan
YES	0.91	122	78	62	65

In Table 5, gamma distribution is better than other ones (In deed, gamma distribution fits the response output best).

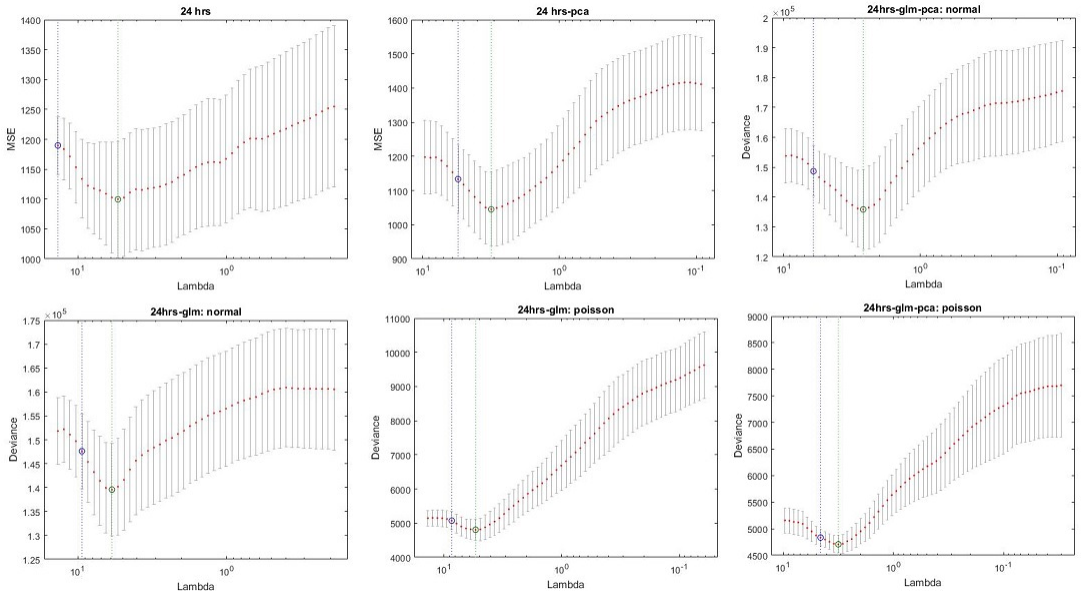

Figure 3: 24 hours

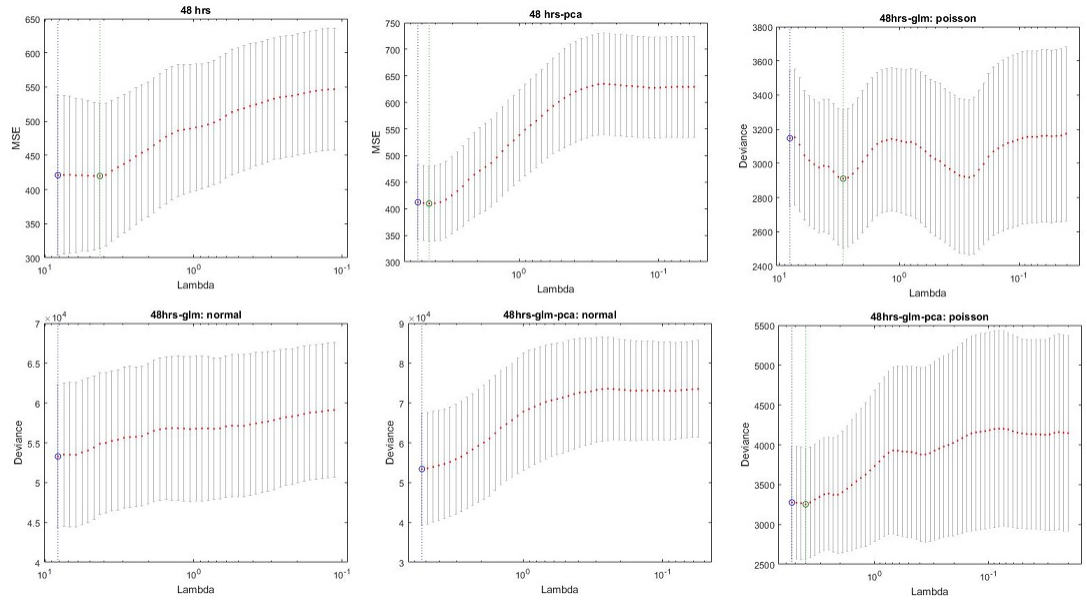

Figure 4: 48 hours

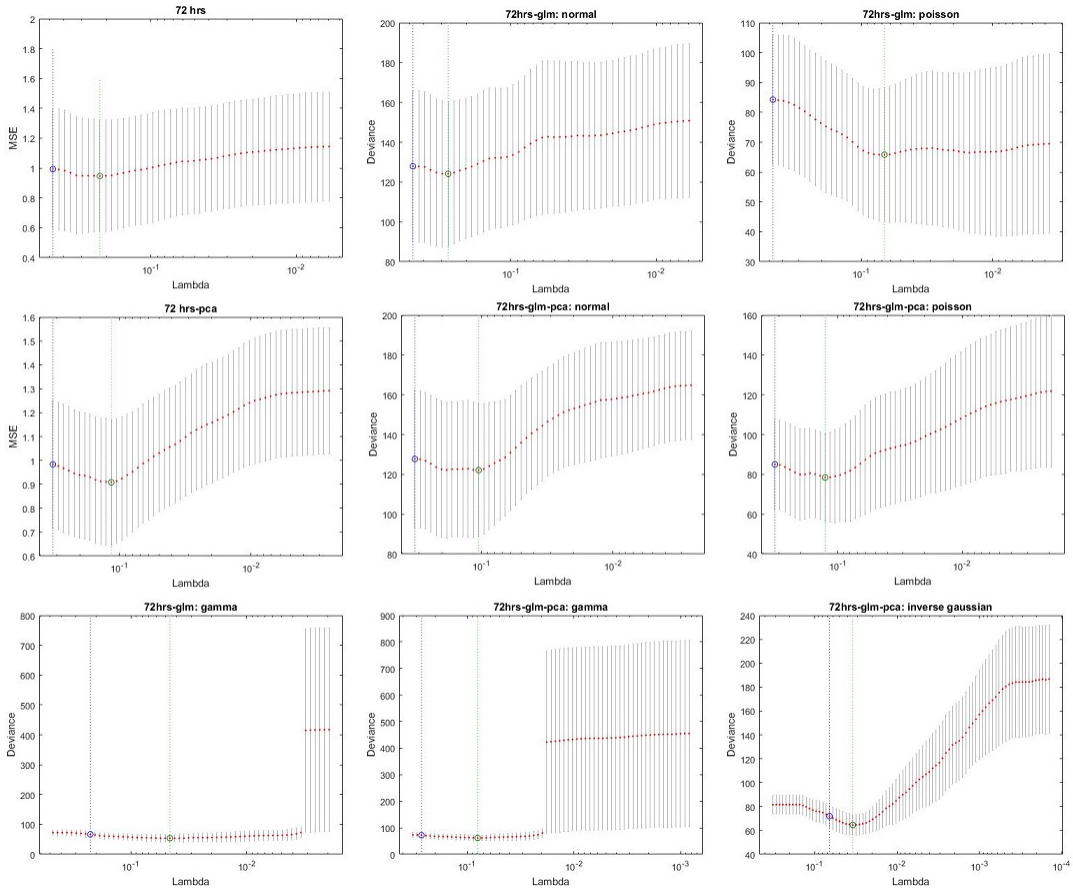

Figure 5: 72 hours

1.4 Conclusion

By the experiment above, we observe that a better distribution fitting to an output, a better lasso (generalized) regression to a dataset. Therefore, we should find an appropriate distribution to an output first, and then apply for corresponding lasso regression.

2 Identification of Raphael's paintings

2.1 The painting set

The data set is provided by Prof. Yang Wang, consisting of 28 paintings which are Raphael's work or not. 21 of them are labeled, 12 positiveness (Raphael's) and 9 negativeness (not Raphael's), and the remaining are dispute. Each painting is digitally saved as 3 or 4 matrices, representing different channels. However, since they are essentially single-color paintings, we convert every painting into a sole gray-scale image/matrix at the very beginning of processing.

Our approaches in the following are originated from Liu, Chan, and Yao [1]. For the sake of convenience, let $\mathcal{T} = \{1, 2, \ldots, 28\}$ be the index set of the paintings, and let $\mathcal{T}_p$ and $\mathcal{T}_n$ be subsets collecting indices for positive and negative samples.

Figure 6: Example paintings in the data set. Left: No. 2, Raphael; Mid: No. 11, not Raphael; Right: No. 10, dispute.

2.2 Feature extraction

To obtain a set of numerical features from the paintings, we adopt the geometric tight frame which has 18 filters as below.

$$\tau_0 = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \quad \tau_1 = \frac{1}{16}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad \tau_2 = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix},$$

$$\tau_3 = \frac{\sqrt{2}}{16}\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}, \quad \tau_4 = \frac{\sqrt{2}}{16}\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}, \quad \tau_5 = \frac{\sqrt{7}}{24}\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix},$$

$$\tau_6 = \frac{1}{48}\begin{bmatrix} -1 & 2 & -1 \\ -2 & 4 & -2 \\ -1 & 2 & -1 \end{bmatrix}, \quad \tau_7 = \frac{1}{48}\begin{bmatrix} -1 & -2 & -1 \\ 2 & 4 & 2 \\ -1 & -2 & -1 \end{bmatrix}, \quad \tau_8 = \frac{1}{12}\begin{bmatrix} 0 & 0 & -1 \\ 0 & 2 & 0 \\ -1 & 0 & 0 \end{bmatrix},$$

$$\tau_9 = \frac{1}{12}\begin{bmatrix} -1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \quad \tau_{10} = \frac{\sqrt{2}}{12}\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \tau_{11} = \frac{\sqrt{2}}{16}\begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & -2 \\ -1 & 0 & 1 \end{bmatrix},$$

$$\tau_{12} = \frac{\sqrt{2}}{16}\begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & -2 & 1 \end{bmatrix}, \quad \tau_{13} = \frac{1}{48}\begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}, \quad \tau_{14} = \frac{\sqrt{2}}{12}\begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\tau_{15} = \frac{\sqrt{2}}{24}\begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \end{bmatrix}, \quad \tau_{16} = \frac{\sqrt{2}}{12}\begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix}, \quad \tau_{17} = \frac{\sqrt{2}}{24}\begin{bmatrix} -1 & 0 & -1 \\ 2 & 0 & 2 \\ -1 & 0 & -1 \end{bmatrix}.$$

[1] Liu, Haixia, Raymond H. Chan, and Yuan Yao. "Geometric tight frame based stylometry for art authentication of van Gogh paintings." Applied and Computational Harmonic Analysis 41.2 (2016): 590-602.

After the application of the tight frame, we obtain coefficient matrices $A^{(i,j)}, j = 0, 2, \ldots, 17$ for every painting $i \in \mathcal{T}$. Then we further exploit the following three statistics of every coefficient matrix as features of a painting.

$$\mu^{(i,j)} \equiv \frac{1}{m_i n_i} \sum_{l=1}^{m_i} \sum_{k=1}^{n_i} A_{l,k}^{(i.j)},$$

the standard deviation

$$\sigma^{(i,j)} \equiv \sqrt{\frac{1}{m_i n_i - 1} \sum_{l=1}^{m_i} \sum_{k=1}^{n_i} \left[A_{lk}^{(i.j)} - \mu^{(i,j)} \right]^2},$$

and the number of nonzeros

$$p^{(i,j)} \equiv \frac{\#\hat{A}^{(i,j)}}{m_i n_i}$$

of the tail matrix

$$\hat{A}_{lk}^{(i,j)} \equiv \begin{cases} A_{lk}^{(i,j)}, & \text{if } |A_{lk}^{(i,j)} - \mu^{(i,j)}| > \sigma^{(i,j)}, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, for painting $i \in \mathcal{T}$, there are totally $18 + 18 + 18 = 54$ feature scores. Let $\mathcal{F} = \{1, 2, \ldots, 54\}$ be the index set of those features.

2.3 Forward dimension reduction: Stage-wise feature selection

We measure the 'score' (i.e. the probability of being positive/genuine) of every painting by the *distance* between them and the sample mean of all genuine paintings in the training set with regard to certain features. More precisely, for the j-th painting and selected features $\mathcal{G} \subset \mathcal{F}$, the score is

$$d_j^{\mathcal{G}} \equiv \| \left(X_{ij}\right)_{j \in \mathcal{G}} - c^{\mathcal{G}} \|_2,$$

where

$$c^{\mathcal{G}} \equiv \frac{1}{|\mathcal{T}_p|} \sum_{i=1}^{|\mathcal{T}_p|} \left(X_{ij}\right)_{j \in \mathcal{G}}.$$

1. Normalize X to make every column of X have unit standard deviation.
2. For current subset $\mathcal{G} \subset \mathcal{F}$ ($\mathcal{G} = \emptyset$ at the beginning), find j^* which solves

   $$\max_{j \in \mathcal{F} - \mathcal{G}} \text{AUC}\left(\mathcal{G} \cup \{j\}\right)$$

   where AUC stands for the *area under the receiver operating characteristic (ROC) curve*.
3. Enlarge $\mathcal{G}$ to include j^* until the certain a set size (i.e. number of features) is achieved.

Note that ROC curve is the graph of true positive rate (TPR) versus false positive rate (FPR). If we starts with a sample who has the least probability of positiveness among the set and adding samples in the order of probability, then as the number of samples increasing, TPR or FPR grows alternately. The case TPR grows faster is favorable, which result in higher AUC.

2.4 Classification criterion

Once we have the scores $d_j^{\mathcal{G}}, j = 1, 2, \ldots, |\mathcal{T}|$ for every sample with respect to a feature subset $\mathcal{G}$, the threshold δ for the classification of positive and negative samples is chosen by maximizing the accuracy i.e. sum of the number of true positive samples ($d_j^{\mathcal{G}} < \delta$ and $j \in \mathcal{T}_p$) and the number of true negative samples ($d_j^{\mathcal{G}} \geq \delta$ and $j \in \mathcal{T}_n$).

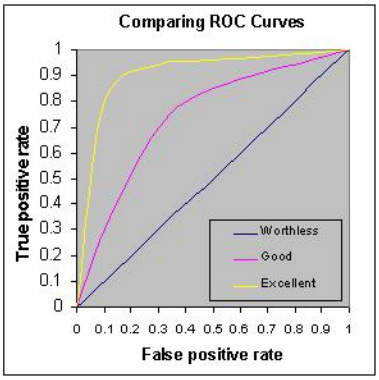

Figure 7: Example ROC curves. (source: `http://gim.unmc.edu/dxtests/roc3.htm`)

2.5 Result and discussion

In our experiment, we perform a *leave-one-out* training and validating strategy. With the 21 labeled paintings, we have 21 pairs of training and test sets and the overall accuracy could be the performance index for the training model.

Unfortunately, we have not received a satisfying accuracy. The predictions are not stable for some left-out paintings, and always failed for several others. The predictions for the dispute paintings are therefore unreliable. As an additional effort, we ever tried some other methods, such as the *support vector machine*, but it led to similar results. It suggests that there are probably some bugs in the code, or a certain preprocessing is necessary.

The source code and latest result (if any) will be updated at

`https://drive.google.com/open?id=0B3zEmYEa2Wc5bXh0YUxuSGJHb2M`

Note that the source data i.e. the painting set is not included because of the property right issue.