MLPR Report

Pengfei Gao, s1144374

Q1.

(a)

The accuracy of the maximum-likelihood estimator is rely on the sufficiently large number of observations. As the training datasets get larger, the likelihood gets smaller and the log scale is better. As here the dataset is small, I suggest to use the k-fold cross-validation method to estimate the two classifiers which all the data used once as the validation data.

(b)

The adaptive radial basis functions give a local models which output is a nonlinear function of the distance of the input from a particular point. Therefore, if the data is far from the trained model, the predication will be very small even zero. Here we want to predicate the extreme cases while the training data is about normal cases, so the adaptive RBF is not good for this situation. I suggest to use multilayer perceptron method which can distinguish data that is not linearly.

(c)

Although multi-layer perceptron method with two hidden layer can approximate any non-linear function, if the data set is complex and many hidden units are used, the training time will become very long and may cause the overfitting problem. Meantime, two hidden layers may introduce a great risk of converging to a local minima. As the problem is never tackled before and know little about it, I suggest to use the logistic regression or PCA to get an overall idea and try to find the best method for this problem.

Q2

(a)

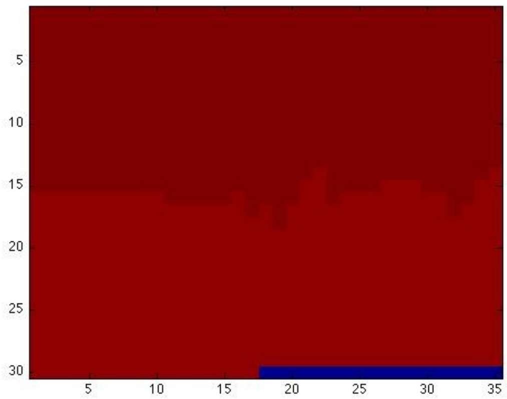


Fig 1. The mean image

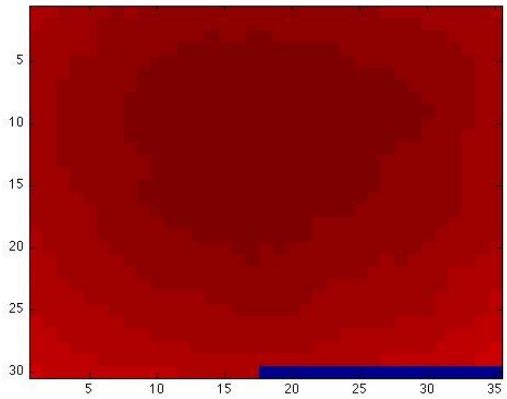


Fig 2. The first eigenvector image

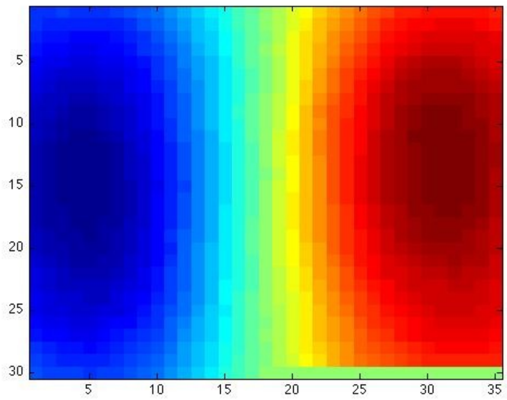


Fig 3. The second eigenvector image

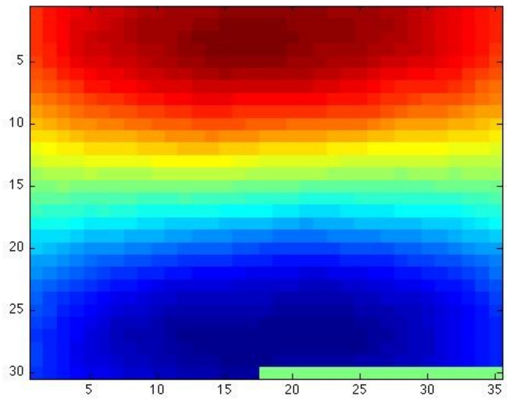


Fig 4. The third eigenvector image

(b)

The number of the worst represented image is 34729.

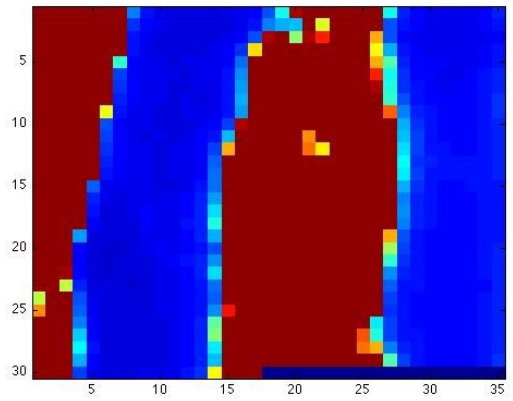


Fig 1. The original image of 34729

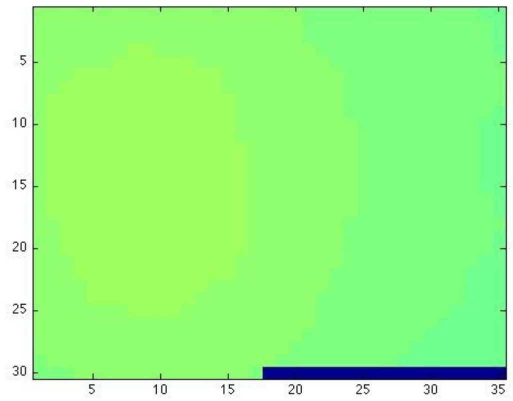


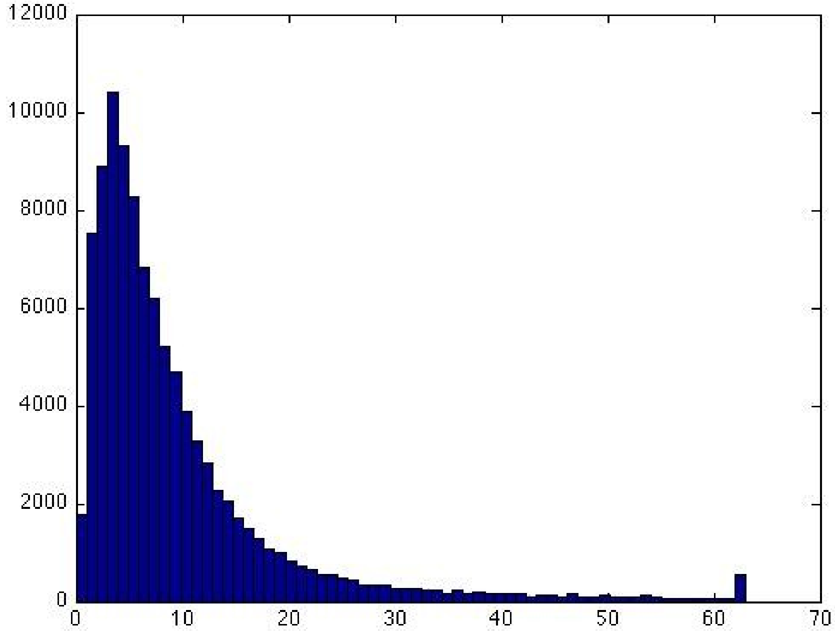
Fig 2. The reconstructed the image

As we can see the images from the original image, the image can be separated into 4 part from left to right horizontally. Each part contains mainly one color, red,green, red, green, respectively. After the 3 dimensional PCA projection, the color of the reconstructed image increases slowly from left to right and radiate from the center which closes to the left side. The reconstructed image can also be separated by 4 parts which look like part of concentric circles.

However, the first eigenvector describes the feature the color radiate from the center with the color value gradually decreases. The second eigenvector describes the feature that the color (intensity around 0) radiates from the left side increasing gradually and the color(intensity around 63) radiates from the right side decreasing gradually and finally they meet at the horizontal middle line of the image. The third eigenvector describes the feature that the color (intensity around 0) radiates from the bottom increasing gradually and the color(intensity around 63) radiates from the upper side decreasing gradually and finally they meet at the vertical middle line of the image.

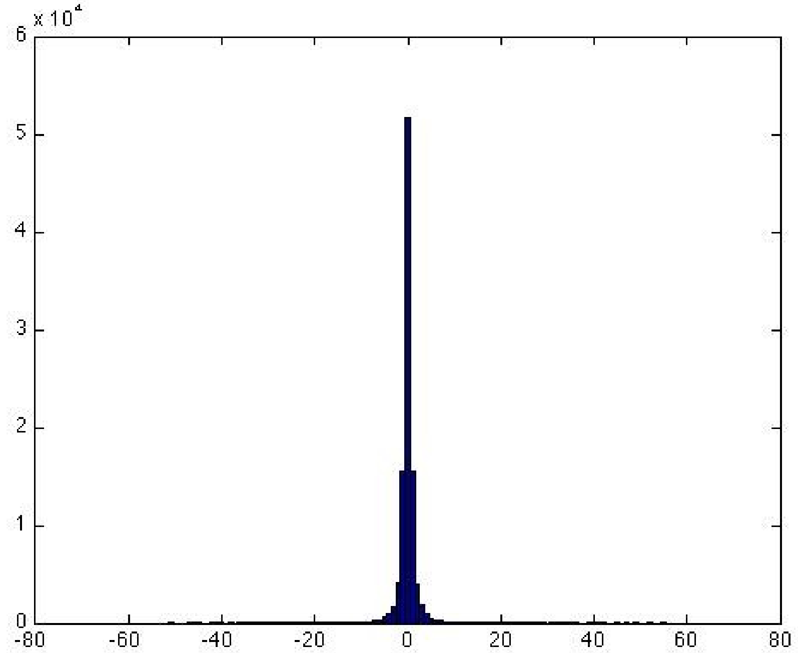
So we can conclude from the above that the original image does not match any features describes by these three eigenvector.

(c)



Lots of the y values are located in the area between 0 to 30. The distribution first increases dramatically from value 0 to 4 and then drop rapidly from 4 to 62. The distribution of the data looks like poisson distribution but he number of the last value 64 is much higher than its neighbors. Similarly the distribution looks like a unimodal but considering the last attributes it actually a bimodal probability distribution.

(d)



51.73% of the training targets are exactly the same as the last element and 15.61%, 15.50% of the training targets is larger than the last element by -1 and 1 respectively. 94.4% of the targets locates in the difference scale [-3,3]. Relating to the previous histogram, we can predict the histogram of the last element is almost the same distribution. This indicates the target value is dependent on the the last element value.

Q3

(a)

We will face the problem that there is too much parameters. As we have 3 features x_1, x_2, x_3 and 64 classes for Y and each $P(x_i|Y=y)$ can be expressed in terms of 64 parameters, the model has $64\times64\times3+64$ parameters. With too many parameters, it is not always possible to such a set of parameter values that uniquely optimizes the log-likelihood. So we need to reduce the numbers of the parameters.

(b)

I.The perplexity is 4.6797.

From the notes, I get Dirichlet distribution:

$$P(x=j|D,\boldsymbol{\theta}_{max},\boldsymbol{\alpha})=\frac{(\alpha_j+c_j)}{(\sum_j(\alpha_j+c_j))}$$

Naive bayes:

$$P(y|x_1,x_2,x_3)=\frac{P(x_1|y)P(x_2|y)P(x_3|y)P(y)}{\sum_{y=0}^{63}P(x_1|y)P(x_2|y)P(x_3|y)P(y)}$$

Finally given the data $\{(\mathbf{x}^n,y^n),n=1,2,\ldots,N\}$, the log likelihood is $\zeta=logP(y^n|D^n)$.

ii.

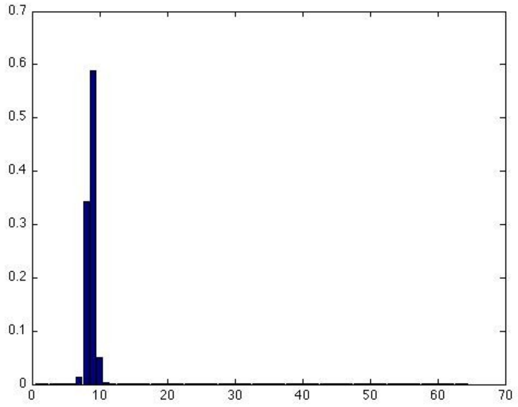


Image 1, Correct prediction, intensity=8

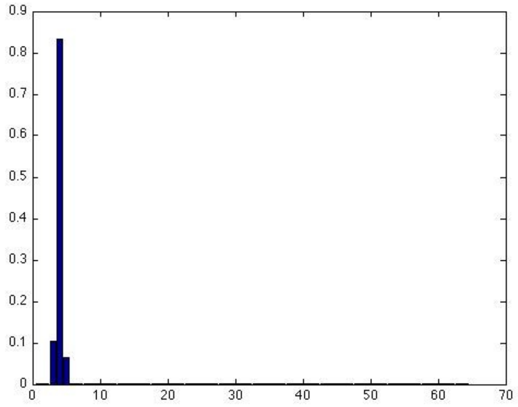


Image 2, Wrong prediction, predicated value=3, real intensity=1

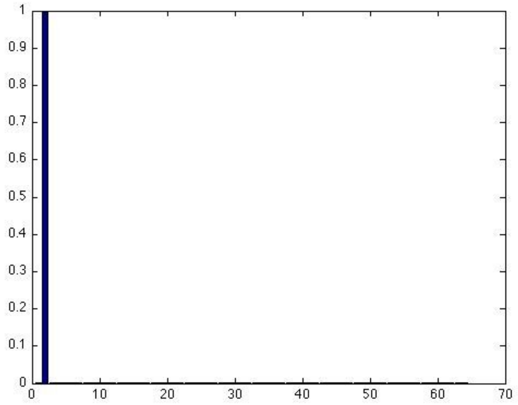


Image 3, Correct prediction, intensity=1

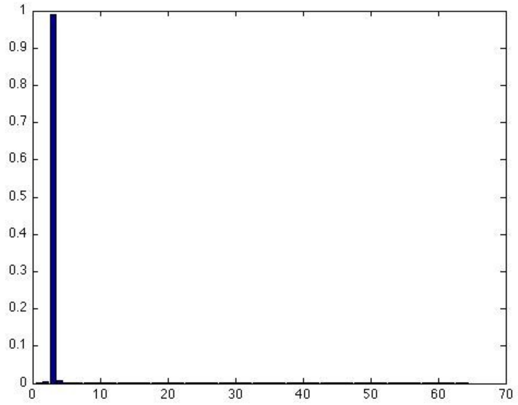


Image 4, Correct prediction, intensity=2

We can see from the above images that most of the prediction cases predicate a probability which is larger than 80% and even closer to 1 for a certain intensity (Image 2,3,4) while other cases may predicate two (or even more) probabilities which are close to each other for several intensities. As (x_1, x_2, x_3) are assumed independent from each other and the training dataset is not sufficient enough, some conditional probabilities $P(x_i|y)$ will be small as the (x_i, y) occurs rarely in the training dataset and some also will be very large as they occur more times. This means this naïve bayes model fails to capture the relations among the (x_1, x_2, x_3, y) . In fact, according to the image pixel kernel convolution, a pixel is affected by its neighbors (e.g. for a 3x3 matrix, the 8 surrounding pixels will affect the central pixel). So we can assume the relationships among (x_1, x_2, x_3, y) is that (x_1, x_2) conditional independent on (x_3) while (y) conditional independent on (x_1, x_2) . As $P(Y|x_1, x_2, x_3) \propto P(Y)P(x_1|Y)P(x_2|Y)P(x_3|Y)$, this will cause the current conditional probabilities inaccurate to describe likelihood which leads to the inappropriate predictive distributions.

iii.

Features are conditional independent is the strong assumption made by the Naive Bayes. However in this case and according to above question answer, (x_1, x_2, x_3) are not completely independent on each other while (x_1, x_2) are conditional independent on x_3 . The assumption here is not perfect as I stated in the above question. But naïve bayes still works very well in this situation comparing to Linear Regression and PCA.

Q4

(a)

The perplexity here is 6.7479.

As the linear regression model is defined as $Y = \mathbf{w}^{\boldsymbol{T}} \phi + \eta$ where $\phi = (1, (\mathbf{x})^{\boldsymbol{T}})^T$. Considering errors in the variables, we add an error to the function $Y = \mathbf{w}^{\boldsymbol{T}} \phi + \eta + \varepsilon$ and $\varepsilon \in N(0, \sigma^2)$. I used the Matlab *mvregress* function to train the dataset. This function returns the biased estimator w and error variance σ^2 .

Assume the errors match the Gaussian distribution, the predicative pdf is $P(Y|\mathbf{w}, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-1}{2\sigma^2}(Y - \mathbf{w}\phi)^2}$. Then given the data $\{(\mathbf{x}^{\boldsymbol{n}}, y^n), n = 1,2,..., N\}$, the log likelihood is $\zeta = logP(y^n|D^n) = \frac{-1}{2\sigma_\eta^2} \sum_{n=1}^{N} (y^n - \mathbf{w}\phi^n)^2 - \frac{N}{2} \log(2\pi\sigma^2)$.

(b)

The perplexity here is 15.6090.

First I calculated the eigenvectors and the mean of the training dataset and then project the training data by the first 10 principal components. For the image i and each component j, use the following function: $pca10d(i,j)=(data(i,:)-mean)\times E(:,j)$, where j is $1:10$ and E is the eigenvector matrix. Then use the $pca10d$ to do linear regression to get the $\boldsymbol{w},\sigma^2$. After this, use the $E, mean$ to project test data on the 10 components using the function: $test10d(i,j)=(testdata(i,:)-mean)\times E(:,j)$. Then do linear regression on the $test10d$ using the $\boldsymbol{w},\sigma^2$ to get the error predicative pdf : $P(Y|\mathbf{w},\sigma^2)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{\frac{-1}{2\sigma^2}(Y-\mathbf{w}\phi)^2}$. Finally given the data $\{(\mathbf{x}^{\boldsymbol{n}},y^n),n=1,2,..,N\}$, the log likelihood is $\zeta=logP(y^n|D^n)=\frac{-1}{2\sigma_\eta^2}\sum_{n=1}^{N}(y^n-\mathbf{w}\phi^n)^2-\frac{N}{2}\log(2\pi\sigma^2)$.

(c)

We have used naïve bayes, linear regression and linear regression on 10 components PCA model. As the perplexity shows, given this data, naïve bayes is quite better than linear regression methods while linear regression is two times better than the PCA.

Naive bayes is a probabilistic method based on applying Bayes's theorem on the conditional independence assumption. We can see from the histogram in Question 2(d), the target value is almost completely conditional independent on each x_i. The amount of training data here is small comparing the testing data which affect the naïve bayes less than LR and PCA. Because we actually estimate those independent distribution as one dimensional distribution. What's more, as the correct prediction probabilities is more probable than the wrong ones, the estimator can still be accurate even the probabilities are not estimated very well. So naïve bayes performs very well in this situation.

Linear regression is a method to modeling the relationship between the target and the data. As image pixels intensities changes gradually and $\{x_1,x_2,x_3\}$ (also shown in histogram in Question 2(d)) are y 's neighbors on its left side, the data is linearly related to the target y. So LR is very good for this model. However, the insufficient training data affects the accuracy of this model. So, the linear regression is less accurate and efficient than the naïve bayes.

PCA is a method trying to reduce the data dimensionality and represent data using principal components which are major features extracted from the whole dataset. The reduction of dimensionality causes that some information which is not uncommon for the whole dataset are removed. Here we use 10 components to represent the data, which is still contains most information about the original data which can be treated as redundant data. This may cause some data become not linearly. So it is much worse than the LR.

Q5

The Bayesian belief network is a directed graph with the associated probability. Features of the data are the nodes in this network and their conditional dependencies are represented through a directed acyclic graph.

According to the image pixel kernel convolution, a pixel is affected by its neighbors (e.g. for a 3x3 matrix, the 8 surrounding pixels will affect the central pixel). Because in normal cases, the color in a image changes gradually which the effects reflect on the pixels is the pixel intensity distribution matches some distribution, e.g. Gaussian distribution, linear regression. In this case, the given data is the upper left corner for the target pixel. This turns out to match the conditional independences in Bayesian belief network.

x_3	x_2
x_1	y

The above table is the location of the (x_1, x_2, x_3, y) in the image. So we can assume the relationships among (x_1, x_2, x_3, y) is that (x_1, x_2) are conditional independent on (x_3) while (y) is conditional independent on (x_1, x_2) .

I estimated the $P(y)$, $P(x_3)$, $P(x_1|y)$, $P(x_2|y)$, $P(x_1|x_3)$ and $P(x_1|x_3)$ by calculating an estimate for the class probability from the training dataset ((prior for a given class) = (number of samples in the class) / (total number of samples)).
The probabilities for various target's value are calculated using the following function:

$$P(y|x_1, x_2) = \frac{P(x_1|y)P(x_2|y)P(y)}{\sum_{x_3} P(x_1|x_3)P(x_2|x_3)P(x_3)}$$

After 4-fold cross-validation, the perplexity I got is 3.7649.