s23 P-set 7 (for your use only, please do not circulate)

1. (a) LS: The fitted model with LS estimated parameters is (partial output omitted; fitted parameter values vary):

$$Y = 2.89 + 11.95X_1 - 10.20X_2 + \epsilon$$

(b) True β's and goodness of the LS estimates: The true parameter values are $\beta_0 = 3, \beta_1 = 1, \beta_2 = 1$.
The LS estimates (and their standard errors) are $\hat{\beta}_0 = 2.9(0.1), \hat{\beta}_1 = 11.9(18.8), \beta_2 = -10.2(18.8)$.
Clearly β_1 and β_2 are not good estimates, indicated by the large standard errors associated with the estimates.
Because of the collinearity in X, the matrix X^TX is near singular (near zero eigenvalue, large condition number). Therefore $(X^TX)^{-1}$ in estimation of β rendered poorly.
However, we can see that approximately, $\hat{\beta}_1 + \hat{\beta}_2 \approx \beta_1 + \beta_2$.

(c) Compare RSS: In this run, $RSS_{LS} = 27.106$, $RSS_{true} = \sum_{i=1}^{30}(Y_i - 3 - x1_i - x2_i)^2 = 29.143$.
The two values are close in value, we even have $RSS_{LS} < RSS_{true}$.
The LS fitted y's are not too far off, even though the individual parameter estimates by LS in (a) are far off. On the other hand, the example indicates that 'good' RSS does not necessarily imply good parameter estimates.

(d) Fit Ridge with $\lambda = 1$: The fitted Ridge model is

$$Y = 2.90 + 0.88X_1 + 0.85X_2 + \epsilon$$

which is quite close to the true model. The parameter estimates are good.

(e) LS method tries to minimize

$$RSS_{LS} = \sum_{j=1}^{n}[y_j - (\beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j})]^2$$

Ridge Regression tries to minimize

$$RSS_{Ridge} = \sum_{j=1}^{n}[y_j - (\beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j})]^2 + \lambda(\beta_1^2 + \beta_2^2)$$

From the results in (a) and (d) we can see that the parameter estimates are much better by Ridge Regression than by Least Square method.
Ridge Regression method has the effect of "shrinking" the magnitude of parameter estimates $\hat{\beta}_i$ (towards 0).

2. Setup training data set, validation set, and input-output variables.

(a) The range of the variables are comparable. We may first consider using the variables without normalization, which has the advantage of easy interpretation of the coefficients.
Fit main effects LASSO regression on the training set:

```
Bfit=glmnet(X,Y)
plot(Bfit,label=T,main="LASSO Boston housing, training (1:300)",cex.main=.8)
```

The coefficient plot shows the order of non-zero coefficients selected as the L_1 norm bound s gets larger ($\sum \|\beta_i\|_1 \leq s$). (plot omitted)
Obtain predictions on testing dataset. Here we use two λ's to obtain two sets of predictions.

```
# Lasso prediction
nX=as.matrix(CVdata[,1:13])
predict(Bfit,newx=nX,s=c(0.1,0.05))
```

To determine the model with optimal λ, we use cross validation to find candidate λ.
Here the cross validation uses the cv.glmnet command on the full dataset. It can be carried out on the training dataset or with your own code.

```
# Lasso cross validation - Which lambda?
Bcvfit=cv.glmnet(X,Y)
plot(Bcvfit)
```

The Mean-Squared Error (MSE) plot gives the candidate λ's (on $\log(\lambda)$ scale) which minimize the MSE or with acceptable small MSE. (plot omitted)

```
Bcvfit$lambda.min  # 0.009823431 (varies for each CV run)
Bcvfit$lambda.1se  # 0.3698459 (varies: 0.3070525, ...)
coef(Bcvfit,s="lambda.min")  # include all covariates
coef(Bcvfit,s="lambda.1se")
14 x 1 sparse Matrix of class "dgCMatrix"
1
(Intercept) -20.251069613
crim          .
zn            .
indus         .
chas          .
nox           .
rm            9.128888114
age          -0.013441770
dis          -0.166737259
rad           .
tax          -0.007896896
ptratio      -0.549668652
black         0.005713891
lstat        -0.126041553
```

The model using $\lambda = 0.37$ (lambda.1se for this cross validation run) has the form

$$medv \sim rm + age + dis + tax + ptratio + black + lstat$$

With $\lambda = 0.37$ (lambda.1se), the fitted model with coefficients is

$$E(medv) = -20.251 + 9.129(rm) - 0.013(age) - 0.167(dis) - 0.008(tax) - 0.550(ptratio) + 0.006(black) - 0.126(lstat)$$

To compare with normalized data:

```
NormX = as.matrix(X)%*%solve(diag(sqrt(diag(var(X)))))
NormBfit=glmnet(NormX,Y)
plot(NormBfit,label=T,main="LASSO Boston housing, training (1:300) normalized",cex.main=.8)
```

The order of variable selection changed some. (plot omitted) The candidate lambdas vary a little. The variable included in the selected models are the same: (output omitted)

```
> coef(NBcvfit,s="lambda.1se")    # normalized
```

For this data set, with or without normalization give similar results. To compare with the ordinary LS fit: (output below for grading, will be omitted in handout)

```
summary(lm(Y~X))
```

The linear model with the coefficients selected by the LASSO model using lambda.1se has very similar estimates: (output omitted)

```
summary(lm(Y~X[,c(6:8,10:13)]))
```

(b) Comments: For this data set with more observations than variables, LASSO regressions obtain consistent results as the least square models. However LASSO also works in the case of more input variables than observations, especially when there are not many non-zero coefficients.

3. Analysis should include PCA vs sparse PCA in significant (non-zero) coefficients, in relation to Left-Right, hi-low frequencies, and variations explained.

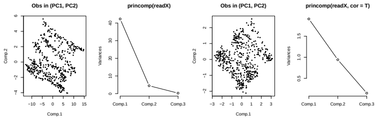
(a) PCA:
Using original data, PC1 loadings are dominated by 4k's, driven by high frequency hearing loss.
Better using scaled data (using cor(data)): First 6 PC carry 96% variation.

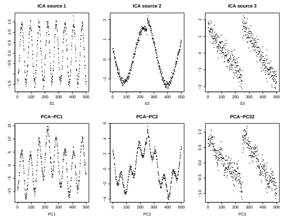
(b) Sparse PCA: Consider up to K=6 PC's. Various settings can be tried to force some PC loadings =0.
SparsePC1 highlights the decibel loss in low frequency, especially on Left ear.
SparsePC2 captures the decibel loss in high frequency, especially on Right ear.
SparsePC3 contrast Left vs Right, mid-frequency. ...
Comment on the variation explained (around 60%)

4. (a) PCA plots (obs on the first two PC's, screeplot): The PCA found the direction of largest variance.
Raw data (left two plots): First two PCs explain 99% of the variation in data.
Var=1 data (right two plots): First two PCs explain 95% of the variation in data.

(b) (ICA, plot components) The top row of the following figures show the independent component recovered by ICA, ordered by the observation numbers. The components (order may differ) are quite non-Gaussian.

(c) (Comparison) The lower row of the above figures show the PCs (ordered by observation number), which look like mixture of the components by ICA. Actually the recovered ICA components are very much like the true inputs, showcase the strength of ICA in separating and recovering independent non-Gaussian data.





5. (a) Derive that univariate normal random variable with mean μ and variance σ^2 has entropy $\log\left(\sigma\sqrt{2\pi e}\right)$.

As derived in class: : Let $\phi(x;\mu,\sigma)$ denote the density function of normal distribution with mean μ and variance σ^2. Then

$$\log\phi(x;\mu,\sigma)=\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right)=\frac{1}{2}\log(2\pi\sigma^2)-\frac{(x-\mu)^2}{2\sigma^2}\log e$$

Using $\int_{\mathbb{R}}\phi(x;\mu,\sigma)dx=1$ and $\int_{\mathbb{R}}(x-\mu)^2\phi(x;\mu,\sigma)dx=\sigma^2$, we have

$$H(X)=-\int_{\mathbb{R}}\phi(x;\mu,\sigma)\log\phi(x;\mu,\sigma)dx=-\int_{\mathbb{R}}\phi(x;\mu,\sigma)\left(-\frac{1}{2}\log(2\pi\sigma^2)-\frac{(x-\mu)^2}{2\sigma^2}\log e\right)dx$$
$$=\frac{1}{2}\log(2\pi\sigma^2)\int_{\mathbb{R}}\phi(x;\mu,\sigma)dx+\frac{1}{2\sigma^2}\log e\int_{\mathbb{R}}(x-\mu)^2\phi(x;\mu,\sigma)dx=\frac{1}{2}\log(2\pi\sigma^2)+\frac{\sigma^2}{2\sigma^2}\log e=\log(\sigma\sqrt{2\pi e})$$

(b) Show $-\int_{\mathbb{R}}f(x)\log\phi(x)dx=\log\left(\sigma\sqrt{2\pi e}\right)$ for $X\sim(0,\sigma^2)$.

Use $\int_{\mathbb{R}}f(x)dx=1$, $\int_{\mathbb{R}}x^2f(x)dx=\sigma^2$, and $\log\phi(x;0,\sigma^2)=-\frac{1}{2}\log(2\pi\sigma^2)-\frac{x^2}{2\sigma^2}\log e$,

$$H(X)=-\int_{\mathbb{R}}f(x)\left(-\frac{1}{2}\log(2\pi\sigma^2)-\frac{x^2}{2\sigma^2}\log e\right)dx=-\frac{1}{2}\log(2\pi\sigma^2)\int_{\mathbb{R}}f(x)dx-\frac{1}{2\sigma^2}\log e\int_{\mathbb{R}}x^2f(x)dx=\log(\sigma\sqrt{2\pi e})$$

(c) Show $H(X)\le H(N(0,\sigma^2))$ for $X\sim(0,\sigma^2)$, equality holds if and only if $X\sim N(0,\sigma^2)$.
(Hint: Use Jensen's Inequality $h(E(X))\le E(h(X))$ for any convex function h and integrable continuous random variable X.)
Apply $-\int_{\mathbb{R}}f(x)\log f(x)dx=-\int_{\mathbb{R}}f(x)\log\phi(x)dx$ obtained in (b),

$$H(X)-H(N(0,\sigma^2))=-\int_{\mathbb{R}}f(x)\log f(x)dx+\int_{\mathbb{R}}\phi(x)\log\phi(x)dx=-\int_{\mathbb{R}}f(x)\log f(x)dx+\int_{\mathbb{R}}f(x)\log\phi(x)dx$$
$$=\int_{\mathbb{R}}f(x)\log\frac{\phi(x)}{f(x)}dx=E_f\left(\log\frac{\phi(x)}{f(x)}\right)\le\log\left[E_f\left(\frac{\phi(x)}{f(x)}\right)\right]=\log\left(\int_{\mathbb{R}}f(x)\frac{\phi(x)}{f(x)}dx\right)=\log\left(\int_{\mathbb{R}}\phi(x)dx)\right)=\log 1=0$$

where $\le$ uses the concavity of log function and Jensen's inequality in the hint. The inequality shows that normal distribution has the largest entropy. The equality $h(E(X))=E(h(X))$ in Jensen's inequality holds when h is linear or when X is a constant. Here $h=\log$, so $\phi(X)/f(X)\equiv c$ for all values of X, or $\phi(x)\equiv cf(x)$. Then $c=1$ since ϕ and f are density functions (≥ 0 and integral to 1). So equality holds if and only if $f=\phi$.

(d) (Maximize entropy for sums of random variables)
For $i=1,2$, let X_i be any continuous random variable on the real line with mean zero and variance σ_i^2. Let $Y=X_1+X_2$. Find $\max_{X_1,X_2}H(Y)$. Describe your choice of X_1,X_2 which yield the maximum entropy of Y.

$$Var(Y)=\sigma^2=\sigma_1^2+\sigma_2^2+2\rho\sigma_1\sigma_2,\qquad \rho=corr(X_1,X_2)$$

From parts (a), (b) and (c), if $X\sim N(0,\sigma^2)$, we always have

$$H(Y)\le H(X)=\log(\sigma\sqrt{2\pi e})=\log\sqrt{\sigma_1^2+\sigma_2^2+2\rho\sigma_1\sigma_2}+\log\sqrt{2\pi e}$$

Choose
$$X_1^*\sim N(0,\sigma_1^2),\quad X_2^*=\frac{\sigma_2}{\sigma_1}X_1^*$$

Then $X_2^*\sim N(0,\sigma_2^2)$. Note that $Var(X_1^*)=\sigma_1^2,\quad Var(X_2^*)=\sigma_2^2,\quad \rho=corr(X_1^*,X_2^*)=1$

Let $$Y^*=X_1^*+X_2^*,\quad then\quad Y^*\sim N(0,\sigma^{*2}),\quad Var(Y^*)=\sigma^{*2}=\sigma_1^2+\sigma_2^2+2\sigma_1\sigma_2$$

and this Y^* achieves the desired maximum entropy:

$$H(Y^*)=\log(\sigma^*\sqrt{2\pi e})=\log\sqrt{\sigma_1^2+\sigma_2^2+2\sigma_1\sigma_2}+\log\sqrt{2\pi e}=\max_{X_1,X_2}H(Y)$$

6. Preparation setup in the EM algorithm for ML of a mixture of multivariate Bernoulli distributions.

$X=[Y_1\ \cdots\ Y_p]'$, where each component Y_i is an independent Bernoulli random variable with parameter ν_i.

(a) $E(Y)=[\nu_1\ \cdots\nu_p]'$ is a p-by-1 vector.

(b) $Cov(Y)$ is a p-by-p diagonal matrix, $Cov(Y)=diag\{\nu_1(1-\nu_1),\cdots,\nu_p(1-\nu_p)\}$.

(c) Now consider Y to be of mixture distribution: with probability π_c, Y is from a p-variate Bernoulli distribution with mean $\boldsymbol{\mu}_c$ and covariance Σ_c, $\sum_{c=1}^K\pi_c=1$. Denote $\boldsymbol{\pi}=[\pi_1\ \cdots\ \pi_K]$.

i. $\boldsymbol{\mu}=E(Y)=\sum_{c=1}^K\pi_c\boldsymbol{\mu}_c=\sum_{c=1}^K\pi_c[\mu_{c1}\ \cdots\ \mu_{cp}]'$.

ii. Write an expression for cond. prob. $P(\boldsymbol{y}|\boldsymbol{\mu}_c)=P(Y=\boldsymbol{y}|\boldsymbol{\mu}_c)$, where $\boldsymbol{y}=[y_1\ \cdots\ y_p]'$ is a realization of Y. Given $\boldsymbol{y}$ is from component c, each independent component $y_i\sim Ber(\mu_{ci})$.

$$P(\boldsymbol{y}|\boldsymbol{\mu}_c)=\prod_{i=1}^p\mu_{ci}^{y_i}(1-\mu_{ci})^{1-y_i}$$

iii. Write an expression for $P(Y=\boldsymbol{y}|\boldsymbol{\pi},\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K)$.
Now $\boldsymbol{y}=[y_1\ \cdots\ y_p]'$ is from a mixture distribution. Denote $\boldsymbol{\mu}_c=[\mu_{c1}\ \cdots\ \mu_{cp}]'$.

$$P(\boldsymbol{x}|\boldsymbol{\pi},\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K)=\sum_{c=1}^K\pi_cP(Y=\boldsymbol{y}|\boldsymbol{\mu}_c)=\sum_{c=1}^K\pi_c\left(\prod_{i=1}^p\mu_{ci}^{y_i}(1-\mu_{ci})^{1-y_i}\right)$$

iv. $$E(Cov(Y|C))=\sum_c\pi_cCov(Y|C=c)=\sum_c\pi_c\Sigma_c,\qquad \Sigma_c=diag\{\mu_{c1}(1-\mu_{c1}),\cdots,\mu_{cp}(1-\mu_{cp})\}$$

We know $E(Y|C=c)=\boldsymbol{\mu}_c$. $E(E(Y|C))=E(Y)=\boldsymbol{\mu}=\sum_{c=1}^p\pi_c\boldsymbol{\mu}_c$,

$$Cov(E(Y|C))=E\left\{[E(Y|C)-E(E(Y|C))]\,[E(Y|C)-E(E(Y|C))]'\right\}=\sum_{c=1}^K\pi_c(\boldsymbol{\mu}_c-\boldsymbol{\mu})(\boldsymbol{\mu}_c-\boldsymbol{\mu})'$$

Thus $$Cov(Y)=E(Cov(Y|C))+Cov(E(Y|C))=\sum_c\pi_c\Sigma_c+\sum_{c=1}^K\pi_c(\boldsymbol{\mu}_c-\boldsymbol{\mu})(\boldsymbol{\mu}_c-\boldsymbol{\mu})'$$

(d) Suppose $\boldsymbol{y}^{(i)}$, $i=1,\cdots,n$ are n independent observations of Y in (c) (which is of the mixture distribution). Each $\boldsymbol{y}^{(i)}=[y_{i1},\cdots,y_{ip}]'$ is from a p-variate Bernoulli distribution with mean $\boldsymbol{\mu}_c$ and covariance Σ_c with probability π_c, $\sum_{c=1}^K\pi_c=1$. Denote the data as $\boldsymbol{Y}=\{\boldsymbol{y}^{(1)},\cdots,\boldsymbol{y}^{(n)}\}$.

i. Write an expression (in terms of y_{ij},π_c,μ_{ci}) for the likelihood function $L\left(\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}|\boldsymbol{Y}\right)=P\left(\boldsymbol{Y}|\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}\right)$.

$$L\left(\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}|\boldsymbol{y}\right)=\prod_{i=1}^nP(\boldsymbol{y}^{(i)}|\boldsymbol{\pi},\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K)=\prod_{i=1}^n\left(\sum_{c=1}^K\pi_cP(\boldsymbol{y}^{(i)}|\boldsymbol{\mu}_c)\right)=\prod_{i=1}^n\left\{\sum_{c=1}^K\pi_c\left[\prod_{j=1}^p\mu_{cj}^{y_{ij}}(1-\mu_{cj})^{1-y_{ij}}\right]\right\}$$

ii. Let the k-vector Z_i be the latent, class membership variable for $\boldsymbol{y}^{(i)}$.
(Recall that a realization $\boldsymbol{z}_i=[z_{i1}\ \cdots\ z_{iK}]'$ has one component $=1$, all other component $=0$.
So $P(Z_i=\boldsymbol{z}_i)=\pi_c$ if and only if $z_{ic}=1,z_{ij}=0$ for $j\neq c$.)
Denote the latent variables as $\boldsymbol{z}=\{\boldsymbol{z}_1,\cdots,\boldsymbol{z}_n\}$.
Write and expression for the full likelihood $P\left(\boldsymbol{Y},\boldsymbol{z}|\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}\right)$ in the form of products.

$$L=P\left(\boldsymbol{Y},\boldsymbol{z}|\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}\right)=\prod_{i=1}^nP\left(\boldsymbol{y}^{(i)},\boldsymbol{z}_i|\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}\right)=\prod_{i=1}^nP\left(\boldsymbol{y}^{(i)}|\boldsymbol{z}_i,\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi}\right)P(Z_i=\boldsymbol{z}_i|\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K,\boldsymbol{\pi})$$
$$=\prod_{i=1}^nP\left(\boldsymbol{y}^{(i)}|\boldsymbol{z}_i,\boldsymbol{\mu}_1,\cdots,\boldsymbol{\mu}_K\right)P(Z_i=\boldsymbol{z}_i|\boldsymbol{\pi})=\prod_{i=1}^n\left\{\prod_{c=1}^K\left(P(\boldsymbol{y}^{(i)}|\boldsymbol{\mu}_c)\right)^{z_c}\right\}\left\{\prod_{c=1}^K\pi_c^{z_c}\right\}=\prod_{i=1}^n\left\{\prod_{c=1}^K\left(\pi_cP(\boldsymbol{y}^{(i)}|\boldsymbol{\mu}_c)\right)^{z_c}\right\}$$
$$=\prod_{i=1}^n\left\{\prod_{c=1}^K\left(\pi_c\prod_{j=1}^p\mu_{ci}^{y_{ij}}(1-\mu_{ci})^{1-y_{ij}}\right)^{z_c}\right\}$$

The log-likelihood in terms of sums is

$$logL=\sum_{i=1}^n\sum_{c=1}^K\log\left(\pi_c\prod_{j=1}^p\mu_{ci}^{y_{ij}}(1-\mu_{ci})^{1-y_{ij}}\right)^{z_c}=\sum_{i=1}^n\sum_{c=1}^K\left(z_c\log\pi_c+z_c\sum_{j=1}^p[y_{ij}\log\mu_{ci}+(1-y_{ij})\log(1-\mu_{ci})]\right)$$