Carnegie Mellon University Department of Statistics & Data Science

36-708 Statistical Machine Learning Homework #3 Solutions

DUE: March 29, 2019

Problem 1 [9 pts.]

Get the iris data. (In R, use `data(iris)`.) There are 150 observations. The outcome is "Species" which has three values. The goal is to predict Species using the four covariates. Compare the following classifiers: (i) LDA **[3 pts.]**, (ii) logistic regression **[3 pts.]**, (iii) nearest neighbors **[3 pts.]**. Note that you will need to figure out a way to deal with three classes when using logistic regression. Explain how you handled this. Summarize your results.

Solution.

In order to deal with multiple classes for logistic regression one can either fit a logistic regression for each of the classes versus the others or use a multinomial logistic regression model. We will use the latter. In these settings the output is a 3 dimensional vector (one per classes) (π_1, π_2, π_3), such that $\sum_i \pi_i = 1$. The model learns β_1 and β_2 such that:

$$\log\left(\frac{\pi_1}{1-\pi_1-\pi_2}\right) = X_i^T\beta_1 \qquad \log\left(\frac{\pi_2}{1-\pi_1-\pi_2}\right) = X_i^T\beta_2$$

We will use the `R` package `nnet` which uses neural networks.

```
set.seed(7)
data(iris)

test_size <- 0.4
test_idx <- sample(nrow(iris), size = as.integer(nrow(iris) * test_size))

X_train <- iris[-test_idx, 1:4]
y_train <- iris[-test_idx, 5]
X_test <- iris[test_idx, 1:4]
y_test <- iris[test_idx, 5]

####### LDA
require(MASS)
fit_lda <- lda(X_train, y_train)
pred_lda <- predict(fit_lda, X_test)
lda_acc <- sum(pred_lda$class == y_test)/(length(y_test))
table(pred_lda$class, y_test)

####### Multinomial Regression
require(nnet)
fit_logreg <- multinom(Species~Sepal.Length +Sepal.Width +Petal.Length +Petal.Width,
              data=iris[-test_idx,])
pred_logreg <- predict(fit_logreg, X_test)
mult_logreg_acc <- sum(pred_logreg == y_test)/(length(y_test))
table(pred_logreg, y_test)
```

```
####### KNN
require(class)
pred_knn <- knn(X_train, X_test, cl=y_train, k=3)
knn_acc <- sum(pred_knn == y_test)/(length(y_test))
table(pred_knn, y_test)
```

LDA and K-nearest neighbors classification achieves 96.67% accuracy, while multinomial logistic regression achieves 95% accuracy. Given the low number of samples, the three models can be considered equivalent in terms of performance over this dataset. A further analysis changing the random split between training and testing set confirms this hypothesis.

Problem 2 [8 pts.]

Use the iris data again but throw away the Species variable. Use k-means^{++} clustering **[4 pts.]** and mean-shift clustering **[4 pts.]**. Compare the clusterings to the true group defined by Species. Which method worked better?

Solution.
We use the R packages and to run k-means^{++} and mean-shift clustering.

```
data(iris)
X <- iris[, 1:4]
y <- iris[, 5]

###### K-MEANS++
library(LICORS)
set.seed(7)
clustering_kmpp <- kmeanspp(X, k = 3, iter.max = 300, nstart = 10)

# Getting the labels of the clustering
table_results_kmpp <- table(clustering_kmpp$cluster, y)
label_kmpp <- apply(table_results_kmpp, 2, which.max)
clustering_labels <- factor(clustering_kmpp$cluster,
                         labels = names(label_kmpp)[order(label_kmpp)])

# Calculating accuracy
acc_kmpp <- sum(y == clustering_labels)/nrow(X)
print(acc_kmpp)

###### Mean Shift
library(LPCM)
set.seed(7)
clustering_ms <- ms(X, h=0.11)

# Getting the labels of the clustering
table_results_ms <- table(clustering_ms$cluster.label, y)
label_ms <- apply(table_results_ms, 2, which.max)
clustering_labels <- factor(clustering_ms$cluster.label,
                         labels = names(label_ms)[order(label_ms)])

# Calculating accuracy
acc_ms <- sum(y == clustering_labels)/nrow(X)
print(acc_ms)
```

As in the mean-shift clustering algorithm we cannot input the number of clusters, the algorithm struggles finding exactly only three clusters. We have used the value $h = 0.11$ for the bandwidth, found via using a validation set, to get exactly three clusters and be able to compare the two methods apple-to-apple. K-means seems to be performing better with an accuracy of 89% against an accuracy of 68%.

Problem 3 [9 pts.]

Download the data from `http://www-bcf.usc.edu/~gareth/ISL/Ch10Ex11.csv`. This is a gene expression dataset. There are 40 tissue samples with measurements on 1,000 genes. The first 20 data points are from healthy people. The second 20 data points are from diseased people.

(a) **[3 pts.]** Use sparse logistic regression to classify the subject. (You may use the function glmnet in R if you like.) Explain how you chose λ. Summarize your findings;

(b) **[3 pts.]** Now use a Sparse Additive Model as described in class. Summarize your findings;

(c) **[3 pts.]** Now suppose we don't know which are healthy and which are diseased. Apply clustering to put the data into two groups. Applying k-means clustering may not work well because the dimension is so high. Instead, you will need to do some sort of dimension reduction or sparse clustering. One very simple method is Sparse Alternate Similarity (arXiv:1602.07277). But you may use any method you like. Describe what you chose to do and what the results are.

Solution.

For sparse logistic regression, we use the `R` package `glmnet`, and we choose the best λ via cross validation. For sparse additive models we use the `R` package `SAM`, in which the run for a series of λ and we select the minimum. For the clustering step we use sparse clustering (Witten, D.M. and Tibshirani, R. *A framework for feature selection in clustering*, 2010) with the `R` package `sparcl`, in which the best bound for the L_1 norm is selected via a permutation approach and the number of clusters is set to 2. All methods perfectly separate the training data. As a note, one could have further split the data into training and testing but, given the low sample size, the most sensible approach would be to use the leave-one-out mis-classification rate as performance metric.

```r
# Read in data
X <- read.csv("http://www-bcf.usc.edu/~gareth/ISL/Ch10Ex11.csv", header=F)

# Generate classes
y <- rep(0,40)
y[21:40] <- 1

#### Sparse Logistic Regression
require(glmnet)

# Tuning Lambda via CV
cvfit <- cv.glmnet(t(X), y, alpha=1, family="binomial", nfold=10)

# Fitting Sparse Logistic Regression
sparse_log_reg <-glmnet(t(X), y, family = "binomial", alpha = 1,
                      lambda = cvfit$lambda.min)
coeff_sparse_log_reg <- as.matrix(coef(sparse_log_reg))
print(nrow(X) - sum(coeff_sparse_log_reg==0)) #Number of >0 coefficients

proba_sparse_log_reg <- predict(object = sparse_log_reg, newx =t(X), type = "response")
pred_sparse_log_reg <- ifelse(proba_sparse_log_reg>0.5, 1, 0)

acc_sparse_log_reg <- sum(y == pred_sparse_log_reg)/length(y)
```

```
print(acc_sparse_log_reg)

#### Sparse Additive Models
require(SAM)

# Fit SAM for 100 lambda - pick the best
sam_model <-samLL(X = as.matrix(t(X)),
                y = as.matrix(y),
                p = 3, nlambda=100)

sam_final <-samLL(X = as.matrix(t(X)),
                y = as.matrix(y),
                lambda =min(sam_model$lambda), p = 3)

# Prediction
pred_sam <-as.data.frame(predict(object = sam_final,
                              newdata = t(X)))

acc_sam <- sum(y == pred_sam)/length(y)
print(acc_sam)

#### Sparse Clustering
library(sparcl)

best_bound_ift <- KMeansSparseCluster.permute(t(x), K=2, wbound=seq(1.1, 100,
    length.out=20))
sparse_cluster <- KMeansSparseCluster(x = t(X), K = 2, wbounds = best_bound_ift$bestw)
acc_sparcl <- sum(y == (as.numeric(sparse_cluster[[1]]$Cs) -1))/length(y)
print(acc_sparcl)
```

Problem 4 [10 pts.]

Let $X = (X_1, \dots, X_d) \in \mathbb{R}^d$. Suppose that $X \sim N(\mu, \Sigma)$. Let $\Omega = \Sigma^{-1}$. Let $j \neq k$ be integers such that $1 \leq j < k \leq d$. Let $Z = (X_s : \ s \neq j, k)$.

(a) **[3 pts.]** Show that the distribution of $(X_j, X_k)|Z$ is $N(a, B)$ and find a and B explicitly.

(b) **[4 pts.]** Show that $X_j \amalg X_k | Z$ if and only if $\Omega_{jk} = 0$.

(c) **[3 pts.]** Now let $X_1, \dots, X_n \sim N(\mu, \Sigma)$. Find the mle $\hat{\Omega}$.

Solution.

(a) More generally, by Theorem 1, for any random vectors $\mathbf{X}_1 \subset X$ and $\mathbf{X}_2 = X \backslash \mathbf{X}_1$, $\mathbf{X}_2 | \mathbf{X}_1$ follows a multivariate normal distribution.

Specifically, if $\mathbf{X}_1 \in \mathbb{R}^r$ and $\mathbf{X}_2 \in \mathbb{R}^s = \mathbb{R}^{d-r}$, then

$$\mathbf{X}_2 | \mathbf{X}_1 = \mathbf{x}_1 \sim N\big(\boldsymbol{\mu}_2 + \Sigma_{21}\Sigma_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\big),$$

where (if necessary) we have reordered X so that

$$X = \left(\begin{array}{c} \mathbf{X}_1 \\ \hline \mathbf{X}_2 \end{array}\right) \quad \text{and} \quad \mu = \mathbb{E}(X) = \left(\begin{array}{c} \boldsymbol{\mu}_1 \\ \hline \boldsymbol{\mu}_2 \end{array}\right) \quad \text{and} \quad \Sigma = \mathrm{Cov}(X) = \left(\begin{array}{c|c} \Sigma_{11} & \Sigma_{12} \\ \hline \Sigma_{21} & \Sigma_{22} \end{array}\right).$$

Proof.

i. $\mathbb{E}(\mathbf{X}_2 | \mathbf{X}_1 = \mathbf{x}_1) = \boldsymbol{\mu}_2 + \Sigma_{21}\Sigma_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1)$

Let

$$\mathbf{A} = \left(\begin{array}{c|c} \boldsymbol{I} & 0 \\ \hline -\Sigma_{21}\Sigma_{11}^{-1} & \boldsymbol{I} \end{array}\right).$$

$\mathbf{A}$ is full-rank so, by Theorem 2,

$$AX = \left(\begin{array}{c} \mathbf{X}_1 \\ \hline \tilde{\mathbf{X}}_2 \end{array}\right)$$

follows a multivariate normal distribution, where

$$\tilde{\mathbf{X}}_2 = \mathbf{X}_2 - \Sigma_{21}\Sigma_{11}^{-1}\mathbf{X}_1.$$

Now,

$$\begin{aligned} \mathrm{Cov}(\mathbf{X}_1, \tilde{\mathbf{X}}_2) &= \mathrm{Cov}(\mathbf{X}_1, \mathbf{X}_2 - \Sigma_{21}\Sigma_{11}^{-1}\mathbf{X}_1) \\ &= \mathrm{Cov}(\mathbf{X}_1, \mathbf{X}_2) - \Sigma_{21}\Sigma_{11}^{-1}\mathrm{Cov}(\mathbf{X}_1, \mathbf{X}_1) \\ &= \Sigma_{12} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{11} \\ &= \Sigma_{12} - \Sigma_{12} \\ &= 0, \end{aligned}$$

(a) so $\mathbf{X}_1$ and $\tilde{\mathbf{X}}_2$ are uncorrelated, and thus independent by Theorem 4. Hence,

$$\begin{aligned}\mathbb{E}(\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1) &= \mathbb{E}(\tilde{\mathbf{X}}_2 + \Sigma_{21}\Sigma_{11}^{-1}\mathbf{X}_1|\mathbf{X}_1 = \mathbf{x}_1)\\ &= \mathbb{E}(\tilde{\mathbf{X}}_2|\mathbf{X}_1 = \mathbf{x}_1) + \Sigma_{21}\Sigma_{11}^{-1}\mathbf{x}_1\\ &= \mathbb{E}(\tilde{\mathbf{X}}_2) + \Sigma_{21}\Sigma_{11}^{-1}\mathbf{x}_1\\ &= \mathbb{E}(\mathbf{X}_2 - \Sigma_{21}\Sigma_{11}^{-1}\mathbf{X}_1) + \Sigma_{21}\Sigma_{11}^{-1}\mathbf{x}_1\\ &= \boldsymbol{\mu}_2 - \Sigma_{21}\Sigma_{11}^{-1}\boldsymbol{\mu}_1 + \Sigma_{21}\Sigma_{11}^{-1}\mathbf{x}_1\\ &= \boldsymbol{\mu}_2 + \Sigma_{21}\Sigma_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1).\ \checkmark\end{aligned}$$

ii. $\mathrm{Cov}(\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1) = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$

Again using the fact that $\mathbf{X}_1$ and $\tilde{\mathbf{X}}_2$ are independent, we have

$$\begin{aligned}&\mathrm{Cov}(\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1)\\ &= \mathrm{Cov}(\tilde{\mathbf{X}}_2 + \underbrace{\Sigma_{21}\Sigma_{11}^{-1}\mathbf{X}_1}_{\text{fixed}}|\mathbf{X}_1 = \mathbf{x}_1)\\ &= \mathrm{Cov}(\tilde{\mathbf{X}}_2|\mathbf{X}_1 = \mathbf{x}_1)\\ &= \mathrm{Cov}(\tilde{\mathbf{X}}_2)\\ &= \mathrm{Cov}(\mathbf{X}_2 - \Sigma_{21}\Sigma_{11}^{-1}\mathbf{X}_1)\\ &= \mathrm{Cov}(\mathbf{X}_2, \mathbf{X}_2) - \Sigma_{21}\Sigma_{11}^{-1}\mathrm{Cov}(\mathbf{X}_1, \mathbf{X}_2) - \mathrm{Cov}(\mathbf{X}_2, \mathbf{X}_1)(\Sigma_{21}\Sigma_{11}^{-1})^T + \Sigma_{21}\Sigma_{11}^{-1}\mathrm{Cov}(\mathbf{X}_1, \mathbf{X}_1)(\Sigma_{21}\Sigma_{11}^{-1})^T\\ &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} - \Sigma_{21}(\Sigma_{11}^{-1})^T\Sigma_{21}^T + \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{11}(\Sigma_{11}^{-1})^T\Sigma_{21}^T\\ &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} + \Sigma_{21}(\Sigma_{11}^T)^{-1}\Sigma_{21}^T\\ &= \Sigma_{22} - 2\Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} + \Sigma_{21}(\Sigma_{11})^{-1}\Sigma_{21}\\ &= \Sigma_{22} - 2\Sigma_{21}\Sigma_{11}^{-1}\Sigma_{21} + \Sigma_{21}(\Sigma_{11})^{-1}\Sigma_{21}\\ &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{21}^T.\ \checkmark\end{aligned}$$

Hence,

$$\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1 \sim N\big(\boldsymbol{\mu}_2 + \Sigma_{21}\Sigma_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\big).$$

Notice that the distribution of $(X_j, X_k)|Z$ is given by the special case where

$$\mathbf{X}_1 = (X_s : s \neq j, k) \quad \text{and} \quad \mathbf{X}_2 = (X_j, X_k).\ \blacksquare$$

(b) As in part (a), let us first reorder X so that

$$X = \left(\begin{array}{c}\mathbf{X}_1\\ \hline \mathbf{X}_2\end{array}\right) \quad \text{and} \quad \Sigma = \mathrm{Cov}(X) = \left(\begin{array}{c|c}\Sigma_{11} & \Sigma_{12}\\ \hline \Sigma_{21} & \Sigma_{22}\end{array}\right),$$

where

$$\mathbf{X}_1 = (X_s : s \neq j, k) \quad \text{and} \quad \mathbf{X}_2 = (X_j, X_k).$$

Similarly, we can partition the (unknown) matrix $\Omega \in \mathbb{R}^{d\times d}$ as

$$\Omega = \left(\begin{array}{c|c} \Omega_{11} & \Omega_{12} \\ \hline \Omega_{21} & \Omega_{22} \end{array}\right),$$

so that

$$\Omega_{22} = \begin{pmatrix} \Omega_{jj} & \Omega_{jk} \\ \Omega_{kj} & \Omega_{kk} \end{pmatrix}.$$

By definition, and using the fact that Σ is symmetric (and thus, so is Ω), we have

$$\Sigma\Omega = \left(\begin{array}{c|c} \Sigma_{11}\Omega_{11} + \Sigma_{12}\Omega_{12}^T & \Sigma_{11}\Omega_{12} + \Sigma_{12}\Omega_{22} \\ \hline \Sigma_{12}^T\Omega_{11} + \Sigma_{22}\Omega_{12}^T & \Sigma_{12}^T\Omega_{12} + \Sigma_{22}\Omega_{22} \end{array}\right) = \left(\begin{array}{c|c} \boldsymbol{I}_{d-2} & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{I}_2 \end{array}\right).$$

Setting each block equal,

$$\begin{aligned} \Sigma_{11}\Omega_{11} + \Sigma_{12}\Omega_{12}^T = \boldsymbol{I}_{d-2} &\implies \Omega_{11} = \Sigma_{11}^{-1} - \Sigma_{11}^{-1}\Sigma_{12}\Omega_{12}^T \\ \Sigma_{12}^T\Omega_{11} + \Sigma_{22}\Omega_{12}^T = \mathbf{0} &\implies \Omega_{12}^T = -\Sigma_{22}^{-1}\Sigma_{12}^T\Omega_{11} \\ \Sigma_{11}\Omega_{12} + \Sigma_{12}\Omega_{22} = \mathbf{0} &\implies \Omega_{12} = -\Sigma_{11}^{-1}\Sigma_{12}\Omega_{22} \qquad (1) \\ \Sigma_{12}^T\Omega_{12} + \Sigma_{22}\Omega_{22} = \boldsymbol{I}_2 &\implies \Omega_{22} = \Sigma_{22}^{-1} - \Sigma_{22}^{-1}\Sigma_{12}^T\Omega_{12}. \qquad (2) \end{aligned}$$

Plugging (1) into (2), we get

$$\begin{aligned} &\Omega_{22} = \Sigma_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12}\Omega_{22} \\ \implies &\boldsymbol{I}_2 = \Sigma_{22}^{-1}\Omega_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12} \\ \implies &(\boldsymbol{I}_2 - \Sigma_{22}^{-1}\Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})\Omega_{22} = \Sigma_{22}^{-1} \\ \implies &(\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})\Omega_{22} = \boldsymbol{I}_2 \\ \implies &\Omega_{22} = (\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})^{-1} \\ \implies &\Omega_{22} = (\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{21}^T)^{-1}. \end{aligned}$$

Using part (a) we see

$$\Omega_{22}^{-1} = \begin{pmatrix} \Omega_{jj} & \Omega_{jk} \\ \Omega_{kj} & \Omega_{kk} \end{pmatrix}^{-1} = \text{Cov}(\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1).$$

" $\implies$ " Suppose $X_j \perp\!\!\!\perp X_k|Z$. Then X_j and X_k are uncorrelated given Z. That is, the off-diagonal elements of the 2×2 matrix $\text{Cov}(\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1)$ are zero. And the inverse of any diagonal matrix is diagonal so $\Omega_{jk} = \Omega_{kj} = 0$.

" $\impliedby$ " Suppose $\Omega_{jk} = 0$. $\Omega^T = (\Sigma^{-1})^T = (\Sigma^T)^{-1} = \Sigma^{-1} = \Omega$, so $\Omega_{kj} = 0$ as well. That is, Ω_{22} is diagonal. Therefore, its inverse $\text{Cov}(\mathbf{X}_2|\mathbf{X}_1 = \mathbf{x}_1)$ is also diagonal, which implies X_j and X_k are uncorrelated given Z. By Theorem 4, X_j and X_k are independent given Z. ∎

(c) The log likelihood in terms of Ω is,

$$l \propto \frac{n}{2}\log|\Omega| - \frac{1}{2}\sum_{i=1}^{n}(x_i-\mu)^T\Omega(x_i-\mu).$$

MLE can be get by taking derivative with respect toΩ and set it to zero,

$$\frac{n}{2}\Omega^{-1} - \frac{1}{2}\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^T = 0,$$

that is,

$$\hat{\Omega} = (\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T)^{-1}.$$

Alternatively, the MLE for Ω, the inverse of Σ, is the inverse of MLE for Σ,

$$\hat{\Omega} = \hat{\Sigma^{-1}} = (\hat{\Sigma})^{-1} = (\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T)^{-1}.$$

Problem 5 [12 pts.]

Let $X = (X_1, X_2, X_3, X_4, X_5)$ be a random vector distributed as $X \sim N(0, \Sigma)$ where

$$\Sigma^{-1} = \begin{pmatrix} 3 & 0 & 1 & 0 & 0 \\ 0 & 3 & 1 & 0 & 0 \\ 1 & 1 & 3 & 1 & 0 \\ 0 & 0 & 1 & 3 & 1 \\ 0 & 0 & 0 & 1 & 3 \end{pmatrix}.$$

(a) **[1 pts.]** What is the graph for X, viewed as an undirected graphical model?

(b) **[2 pts.]** List the maximal cliques of the graph.

(c) **[4 pts.]** Which of the following independence statements are true?

  (a) $X_2 \perp\!\!\!\perp X_3 | X_1, X_2$

  (b) $X_3 \perp\!\!\!\perp X_4 | X_5$

  (c) $\{X_1, X_2\} \perp\!\!\!\perp X_3 | X_4, X_5$

  (d) $X_1 \perp\!\!\!\perp X_5 | X_3$

(d) **[2 pts.]** List the local Markov properties for this graphical model.

(e) **[3 pts.]** Simulate 100 observations from this model. Construct a graph using hypothesis testing. Report your results. Include your code.

Solution.

(a) The edges can be seen directly from Σ^{-1}. That is,

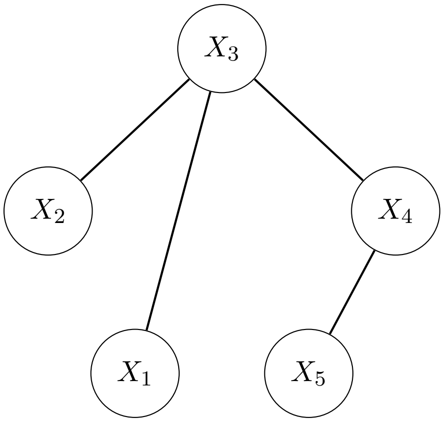


Figure 1. Conditional independence graph of $X = (X_1, X_2, X_3, X_4, X_5)$

(b) By definition, the maximal cliques are $\{1, 3\}, \{2, 3\}, \{3, 4\}, \{4, 5\}$.

(c) (Statement (a) has an typo and is ignored)

These statements can be verified or falsified by simply considering the graph in part (a). To see the conditional independence between set of nodes A and B conditioning on C, check if there is a connecting path between A and B when the nodes in C are blocked. As a result,

  (b) FALSE;
  (c) FALSE;
  (d) TRUE.

(d) Local Markov property is :

$$\forall s \in V,\ p(x_s|x_t,\ t \neq s) = p(x_s|x_t,\ t \in N(s)).$$

Hence, in our case, Local Markov properties is as below :

$$\begin{aligned}
X_1|X_2, X_3, X_4, X_5 &\overset{d}{=} X_1|X_3 \\
X_2|X_1, X_3, X_4, X_5 &\overset{d}{=} X_2|X_3 \\
X_3|X_1, X_2, X_4, X_5 &\overset{d}{=} X_3|X_1, X_2, X_4 \\
X_4|X_1, X_2, X_3, X_5 &\overset{d}{=} X_4|X_3, X_5 \\
X_5|X_1, X_2, X_3, X_4 &\overset{d}{=} X_5|X_4.
\end{aligned}$$

(e) As provided in the lecture notes, you can either construct a marginal correlation graph or a partial correlation graph. We present the code for a partial correlation graph following the normal approximation testing described in page 11. The produced graph is consistent with the graph in part (a).

```
library(MASS)
#generate sample
d = 5; n = 100; alpha = 0.05; m = d*(d-1)/2
omega <- matrix(c(3,0,1,0,0,
                0,3,1,0,0,
                1,1,3,1,0,
                0,0,1,3,1,
                0,0,0,1,3), ncol = 5, byrow = TRUE)
X <-mvrnorm(n, mu = rep(0,d), Sigma = solve(omega))

#estimate matrix R
S_n <- 1/n * t(X)%*%X
hatOmega <- solve(S_n)
Rmat <- -hatOmega/sqrt(outer(diag(hatOmega), diag(hatOmega)))

#test edge
Z <- 1/2*log((1+Rmat)/(1-Rmat))
edge <- abs(Z) > (qnorm(1 - alpha/(2*m))/sqrt(n - d - 1))
```

Problem 6 [8 pts.]

Let $X = (X_1, \dots, X_4)$ where each variable is binary. Suppose the probability function is

$$\log p(x) = \psi_\varnothing + \psi_1(x_1) + \psi_{12}(x_1, x_2) + \psi_{13}(x_1, x_3) + \psi_{24}(x_2, x_4) + \psi_{34}(x_3, x_4).$$

(a) **[3 pts.]** Draw the implied graph;

(b) **[3 pts.]** Write down all the independence and conditional independence relations implies by the graph;

(c) **[2 pts.]** Is the model graphical? Is the model hierarchical?

Solution.

(a) The implied graph is

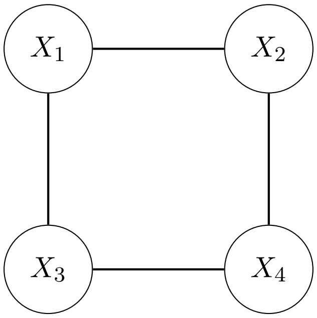


Figure 2. Implied graph of $X = (X_1, X_2, X_3, X_4)$

(b) From Theorem 9 in lectures notes, we have $X_1 \perp\!\!\!\perp X_4 | X_2, X_3$ and $X_2 \perp\!\!\!\perp X_3 | X_1, X_4$.

(c) Not graphical, not hierarchical. This model satisfies $\psi_1(x_1) = 0$ but $\psi_{12}(x_1, x_2) \neq 0$, so the model is not hierarchical. And hence this model is not graphical as well, by Lemma 10 in the Graphical Models lecture notes.

Problem 7 [8 pts.]

Let $X_1, \dots, X_4$ be binary. Draw the independence graphs corresponding to the following log-linear models (where $\alpha \in \mathbb{R}$). Also, identify whether each is graphical and/or hierarchical (or neither).

(a) **[2 pts.]** $\log p(x) = \alpha + 11x_1 + 2x_2 + 3x_3$

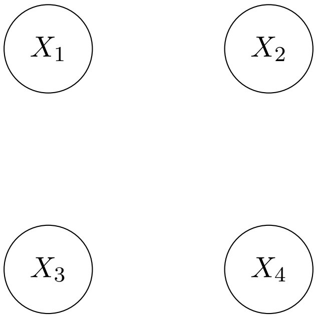
(b) **[2 pts.]** $\log p(x) = \alpha + 2x_2 + 1.5x_3 + 17x_4 + 12x_2x_3 + 78x_2x_4 + 3x_3x_4 + 32x_2x_3x_4$

(c) **[2 pts.]** $\log p(x) = \alpha + 9x_1 + 2x_2 + 1.5x_3 + 17x_4 + 12x_2x_3 + 3x_3x_4 + x_1x_4 + 2x_1x_2$

(d) **[2 pts.]** $\log p(x) = \alpha + 115x_1x_2x_3x_4$.

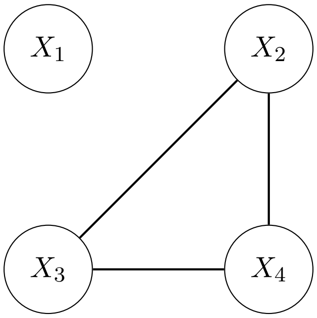
Solution.

(a) Hierarchical, but not graphical.

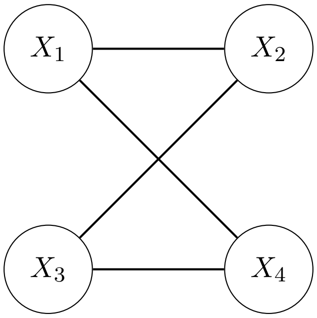


Note: X_4 is a clique, but $\beta_4 = 0$.

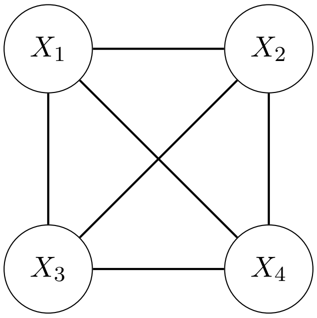
(b) Hierarchical, but not graphical.



(c) Graphical and hierarchical.



(d) Not graphical, nor hierarchical.



Problem 8 [10 pts.]

Consider the log-linear model

$$\log p(x) = \beta_0 + x_1 x_2 + x_2 x_3 + x_3 x_4.$$

Simulate $n = 1000$ random vectors from this distribution. (Show your code.) Fit the model

$$\log p(x) = \beta_0 + \sum_j \beta_j x_j + \sum_{k<\ell} \beta_{k\ell} x_k x_\ell$$

using maximum likelihood. Report your estimators. Use hypothesis testing to decide which parameters are non-zero. Compare the selected model to the true model.

Solution.

To simulate random vectors from the log-linear distribution we simply convert to the corresponding multinomial and sample from it. We then fit the requested model using `glm` command with a log-link function (`family = "poisson"`). The results for $n = 1000$ are given in Table I.

```
# Setup
set.seed(1)
n <- 1000
x1<-x2<-x3<-x4<-seq(0,1,by=1)
grid <- expand.grid(x1,x2,x3,x4)

# Calculate probabilities
p <- rep(NA,16)
for (itr in 1:16){
  p[itr] <- exp(grid[itr,1]*grid[itr,2]+grid[itr,2]*grid[itr,3]+grid[itr,3]*grid[itr,4])
}

# Calculate intercept and adjust probabilities
beta_0 <-log(1/sum(p))
for (itr in 1:16){
  p[itr] <- p[itr] * exp(beta_0)
}

# Sample data and fit GLM model
samp <- sample(1:16,prob=p,size=n,replace=TRUE)
count <- rep(NA,16)
for (itr in 1:16){
  count[itr] <- length(which(samp==itr))
}
data <- cbind(count,grid)
names(data)[2:5] <- c("X1","X2","X3","X4")
model <- glm(count ~ X1 + X2 + X3 + X4 + .*., data = data, family = "poisson")
summary(model)
```

The features deemed significant by a t-test are marked with ***. The results are consistent with the true model.

Table I. Regression summary

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	2.97703	0.17268	17.241	< 2e-16 ***
X1	-0.09754	0.19408	-0.503	0.615
X2	-0.25170	0.20105	-1.252	0.211
X3	0.16575	0.19517	0.849	0.396
X4	-0.09327	0.19281	-0.484	0.629
X1:X2	0.96414	0.16313	5.910	3.42e-09 ***
X1:X3	-0.13305	0.17915	-0.743	0.458
X1:X4	0.14144	0.14858	0.952	0.341
X2:X3	1.13321	0.18387	6.163	7.13e-10 ***
X2:X4	0.21445	0.17296	1.240	0.215
X3:X4	0.84883	0.16996	4.994	5.90e-07 ***

Problem 9 [10 pts.]

Let $X_1, \dots, X_n \in \mathbb{R}^d$. Let Σ be the $d \times d$ covariance matrix for X_i. The covariance graph G puts an edge between (j, k) if $\Sigma_{jk} \neq 0$. Here we will us the bootstrap to estimate the covariance graph.

Let Σ have the following form: $\Sigma_{jj} = 1$, $\Sigma_{j,k} = a$ if $|j - k| = 1$ and $\Sigma_{j,k} = 0$ otherwise. Here, $a = 1/4$.

Let $d = 100$ and $n = 50$. Generate n observations. Compute a 95 percent bootstrap confidence set for Σ using the bootstrap distribution

$$\mathbb{P}\Big(\max_{j,k} \sqrt{n}|\hat{\Sigma}^*_{jk} - \hat{\Sigma}_{jk}| \le t \;\Big|\; X_1, \dots, X_n\Big).$$

This gives (uniform) confidence intervals for all the elements of Σ_{jk}. For each (j, k), put an edge if the confidence interval for Σ_{jk} excludes 0. Plot your graph. Try this for different values of a. Summarize your results.

Solution.

An outline on how to approach this problem is given at page 8 of the Graphical Models notes. In this solution we will use the `R` package `igraph` to visualize the covariance graph after the estimate.

```
require(mvnfast)
require(igraph)

set.seed(7)

# Generate Data
n <- 50
d <- 100
sigma_mat <- toeplitz(c(1, 1/4, numeric(d-2)))
data <- rmvn(n, numeric(d), sigma_mat)
corr_data <- cor(data)

# Run Bootstrap
bootstrap_rep <- 1e+03
stats_boot <- numeric(bootstrap_rep)
for (b in 1:bootstrap_rep) {
  data_boot <- data[sample(1:n, replace=TRUE),]
  corr_data_boot <- cor(data_boot)
  stats_boot[b] =max(abs(corr_data_boot-corr_data))
}

# Calculate Confidence Sets
alpha_cutoff <- quantile(stats_boot,c(0.05))
corr_low <- corr_data - alpha_cutoff
corr_up <- corr_data + alpha_cutoff

# Remove Self-loops
adjacency_mat <- (corr_low > 0| corr_up < 0)
for(i in 1:d){
  adjacency_mat[i,i] = 0
}

# Calculate how many are correctly and wrongly recovered
adj_indeces <-which(adjacency_mat==1, arr.ind = TRUE)
adj_diff_abs <- abs(apply(X=adj_ones, MARGIN = 1, FUN = diff))
sum(adj_diff_abs == 1)
sum(adj_diff_abs > 1)
```

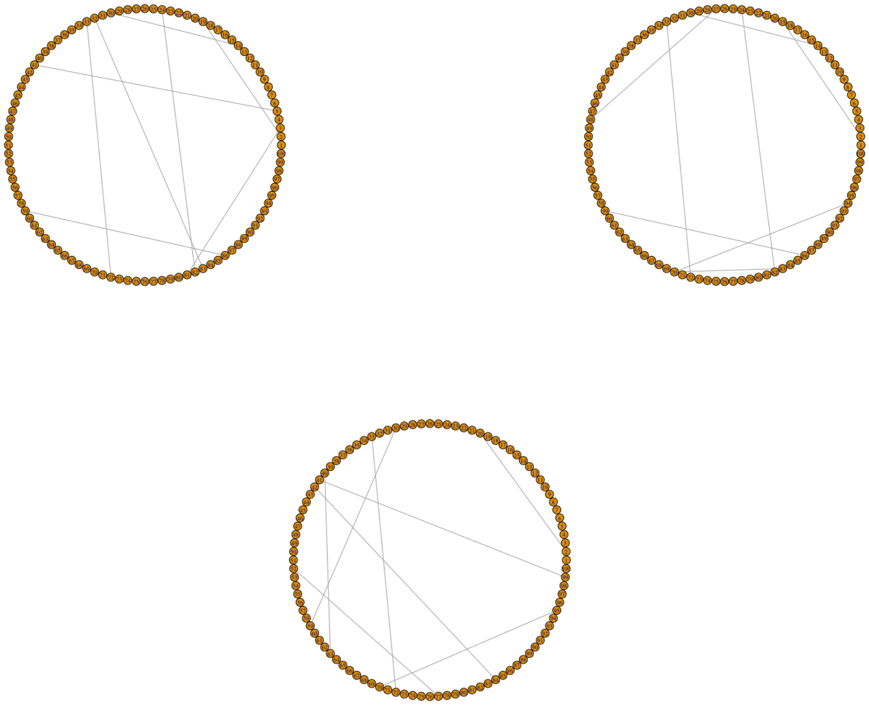


Figure 1: Covariance graph with $a = 1/8, 1/4$ and $1/2$ from top left to center bottom.

```
# Plotting results
colnames(adjacency_mat) <- rownames(adjacency_mat) <- 1:d
graph <- graph_from_adjacency_matrix(adjacency_mat, mode = "undirected", diag = FALSE)
plot(graph, layout = layout.circle, vertex.size=6, vertex.label.cex=0.6)
```

With $a = 1/4$, we recover only 10 out of the of 198 non-zero correlations - excluding the diagonal entries - and wrongly recovering 18 of them. When $a = 1/8$ the number of correctly recovered drops to 2, with the wrongly recovered remaining at 16. When further simulating with different values of lower a, reducing a seems to be in general leading to worse performance. When $a = 1/2$ the number of correctly recovered is 150, with the number of mis-recovered equal to 22. Again using simulations with values of a close to 1/2, recovery performance seem to be improving in this case. It has to be noted that a cannot be larger than 1/2 as the covariance matrix in that case would not be positive definite anymore.

Problem 10 [8 pts.]

Let $A \in \{0,1\}$ be a binary treatment variable and let $Y \in \mathbb{R}$ be the response variable. Let $(Y(0), Y(1))$ be the counterfactual variables where $Y = AY(1) + (1-A)Y(0)$. Assume that

$$Y = \alpha + \gamma A + \sum_{j=1}^{d} \beta_j X_j + \epsilon$$

where $(X_1, \dots, X_d)$ are confounding variables and $\mathbb{E}[\epsilon | X_1, \dots, X_d] = 0$. Assume there are no unmeasured variables.

(a) **(4 pts.)** Let $\theta = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)]$. Show that $\theta = \gamma$.

(b) **(4 pts.)** Suppose now that we do not observe the confounding variables X_j. All we observe is $(A_1, Y_1), \dots, (A_n, Y_n)$. Suppose, unaware of the confounding variables, we fit the linear model $Y = \alpha + \rho A + \delta$ where $\mathbb{E}[\delta] = 0$. Let $\widehat{\rho}$ be the least squares estimator. Show that $\widehat{\rho} \xrightarrow{P} \gamma + \Delta$ for some Δ. Find an explicit expression for Δ.

Solution.

(a)

$$\begin{aligned}
\theta &= \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] \\
&= \mathbb{E}[\mathbb{E}[Y(1)|X_1, \dots, X_n]] - \mathbb{E}[\mathbb{E}[Y(0)|X_1, \dots, X_n]] \\
&= \mathbb{E}\left[\mathbb{E}\left[\alpha + \gamma + \sum_{j=1}^{d} \beta_j X_j + \epsilon \;\middle|\; X_1, \dots, X_n\right]\right] - \mathbb{E}\left[\mathbb{E}\left[\alpha + \sum_{j=1}^{d} \beta_j X_j + \epsilon \;\middle|\; X_1, \dots, X_n\right]\right] \\
&= \mathbb{E}\left[\alpha + \gamma + \sum_{j=1}^{d} \beta_j X_j + \mathbb{E}\big[\epsilon \mid X_1, \dots, X_n\big]\right] - \mathbb{E}\left[\alpha + \sum_{j=1}^{d} \beta_j X_j + \mathbb{E}\big[\epsilon \mid X_1, \dots, X_n\big]\right] \\
&= \alpha + \gamma + \sum_{j=1}^{d} \beta_j X_j - \Big(\alpha + \sum_{j=1}^{d} \beta_j X_j\Big) \\
&= \gamma
\end{aligned}$$

(b)

$$\begin{aligned}
\widehat{\rho} &= \frac{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})(A_i - \overline{A})}{\frac{1}{n}\sum_{i=1}^{n}(A_i - \overline{A})^2} \\
&\xrightarrow{P} \frac{\text{Cov}(Y, A)}{\text{Var}(A)} && \text{WLLN + conv. thm} \\
&= \frac{\text{Cov}(\alpha + \gamma A + \sum_{j=1}^{d} \beta_j X_j + \epsilon, A)}{\text{Var}(A)} \\
&= \frac{\gamma \text{Var}(A) + \sum_{j=1}^{d} \beta_j \text{Cov}(X_j, A) + \text{Cov}(\epsilon, A)}{\text{Var}(A)} \\
&= \gamma + \frac{\sum_{j=1}^{d} \beta_j \text{Cov}(X_j, A) + \text{Cov}(\epsilon, A)}{\text{Var}(A)}
\end{aligned}$$

If we assume $\text{Cov}(\epsilon, A) = 0$ then

$$\Delta = \frac{\sum_{j=1}^{d} \beta_j \text{Cov}(X_j, A)}{\text{Var}(A)}.$$

Problem 11 [8 pts.]

Consider a sequence of time ordered random variables

$$X_1, A_1, Y_1, X_2, A_2, Y_2, X_3, A_3, Y_3, \dots, X_T, A_T, Y_T.$$

Here, the $X_j's$ are the covariates, the $A_j's$ are binary treatment variables and the $Y_j's$ are the response of interest. Assume there are no unobserved confounding variables. The DAG for this model has all directed arrows from the past into the future. That is, the parents for each variables are all variables in its past, For example, the parent of A_1 is X_1. The parents of Y_1 are (X_1, A_1). The parents of X_2 are (X_1, A_1, Y_1) and so on. Let p denote the joint density of all these variables.

(a) **(5 pts.)** Find an explicit expression (in terms of p) for

$$\mathbb{E}[Y_T | A_1 = a_1, \dots, A_T = a_T].$$

(b) **(3 pts.)** Find an explicit expression (in terms of p) for

$$\mathbb{E}[Y_T | \text{ set } (A_1 = a_1, A_2 = a_2, \dots, A_T = a_T)].$$

Solution.

(a) Let $\text{par}(x_j)$ denote the set of parents of X_j on the DAG, and so on.

$$\begin{aligned}
\mathbb{E}[Y_T | A_1 = a_1, \dots, A_T = a_T] &= \int y_T p(y_T | A_1 = a_1, \dots, A_T = a_T) dy_T \\
&= \int y_T \frac{p(y_T, a_1, \dots, a_T)}{p(a_1, \dots, a_T)} dy_T \\
&= \int y_T \frac{\int \cdots \int p(x_1, a_1, y_1, \dots, x_T, a_T, y_T) dx_1 \cdots dx_T dy_1 \cdots dy_{T-1}}{\int \cdots \int p(x_1, a_1, y_1, \dots, x_T, a_T, y_T) dx_1 \cdots dx_T dy_1 \cdots dy_T} dy_T \\
&= \int y_T \frac{\int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(a_j | \text{par}(a_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_{T-1}}{\int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(a_j | \text{par}(a_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_T} dy_T \\
&= \frac{\int y_T \int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(a_j | \text{par}(a_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_{T-1} dy_T}{\int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(a_j | \text{par}(a_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_T}
\end{aligned}$$

(b) Using the expression from part (a), we can *set* $A_1 = a_1, A_2 = a_2, \dots, A_T = a_T$ by replacing $p(a_j | \text{par}(a_j))$ for all $j = 1, \dots, T$ with 1. That is,

$$\begin{aligned}
\mathbb{E}[Y_T | \text{ set } (A_1 = a_1, A_2 = a_2, \dots, A_T = a_T)] &= \frac{\int y_T \int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_{T-1} dy_T}{\underbrace{\int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_T}_{=1}} \\
&= \int y_T \Big(\int \cdots \int \prod_j p(x_j | \text{par}(x_j)) p(y_j | \text{par}(y_j)) dx_1 \cdots dx_T dy_1 \cdots dy_{T-1} \Big) dy_T
\end{aligned}$$

Appendix

Theorem 1 *Suppose that $X = (X(1),\dots,X(d)) \sim N_d(\mu,\Sigma)$. For any $X_1 \subset X$ and $X_2 = X\backslash X_1$, $X_2|X_1$ follows a multivariate normal distribution.*

Proof. We were told we can take this theorem for granted. The parameters that characterize this multivariate normal distribution are computed in Problem 3(a).

Theorem 2 *Suppose that $X = (X(1),\dots,X(d)) \sim N_d(\mu,\Sigma)$. For any full-rank $\mathbf{A} \in \mathbb{R}^{m\times d}$ and $\mathbf{b} \in \mathbb{R}^m$,*

$$\mathbf{A}X + \mathbf{b} \sim N_m(\mathbf{A}\mu + \mathbf{b}, \mathbf{A}\Sigma\mathbf{A}^T).$$

Proof. We use moment generating functions. Let $Y = \mathbf{A}X + \mathbf{b}$. The joint moment generating function of X is

$$M_X(t) = \exp\Big(t^T\mu + \frac{1}{2}t^T\Sigma t\Big).$$

Then the joint moment generating function of Y is

$$\begin{aligned} M_Y(t) &= \exp(t^T\mathbf{b})M_X(\mathbf{A}^Tt) \\ &= \exp(t^T\mathbf{b})\exp\Big(t^T\mathbf{A}\mu + \frac{1}{2}t^T\mathbf{A}\Sigma\mathbf{A}^Tt\Big) \\ &= \exp\Big(t^T(\mathbf{A}\mu + \mathbf{b}) + \frac{1}{2}t^T\mathbf{A}\Sigma\mathbf{A}^Tt\Big), \end{aligned}$$

which is the moment generating function of the joint multivariate normal distribution

$$N_m(\mathbf{A}\mu + \mathbf{b}, \mathbf{A}\Sigma\mathbf{A}^T). \ \blacksquare$$

Corollary 3 *Suppose that $X = (X(1),\dots,X(d)) \sim N_d(\mu,\Sigma)$. Then any p-dimensional subset $\tilde{X}$ of X follows a multivariate normal distribution*

$$\tilde{X} \sim N_p(\tilde{\mu},\tilde{\Sigma}),$$

where $\tilde{\mu}$ is the vector of means of the variables in $\tilde{X} \subseteq X$ and $\tilde{\Sigma}$ is the sub-matrix of Σ obtained by deleting the rows and columns corresponding to the variables in $X\backslash\tilde{X}$.

Theorem 4 *Suppose that $X = (X(1),\dots,X(d)) \sim N_d(\mu,\Sigma)$. Two random-vectors $\tilde{\mathbf{X}}_1 \subset X$ and $\tilde{\mathbf{X}}_2 \subset X$ are independent if and only if they are uncorrelated.*

Proof.
"$\Longrightarrow$" This is true regardless of the distribution. See [?].
"$\Longleftarrow$" By Corollary 2, $\tilde{X} = (\tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2) \in \mathbb{R}^q = \mathbb{R}^{r+s}$ follows a multivariate normal distribution with density

$$f_{\tilde{X}}(\tilde{x}_1,\dots,\tilde{x}_q) = \frac{1}{\sqrt{(2\pi)^q|\tilde{\Sigma}|}}\exp\Big(-\frac{1}{2}(\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})^T\tilde{\Sigma}^{-1}(\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})\Big), \tag{3}$$

with

$$\tilde{\boldsymbol{\mu}} = \left(\begin{array}{c} \tilde{\boldsymbol{\mu}}_1 \\ \hline \tilde{\boldsymbol{\mu}}_2 \end{array}\right) \quad \text{and} \quad \tilde{\Sigma} = \left(\begin{array}{c|c} \tilde{\Sigma}_{11} & 0 \\ \hline 0 & \tilde{\Sigma}_{22} \end{array}\right),$$

where

$$\tilde{\mathbf{X}}_1 \sim N_r(\tilde{\boldsymbol{\mu}}_1, \tilde{\Sigma}_{11}) \quad \text{and} \quad \tilde{\mathbf{X}}_2 \sim N_s(\tilde{\boldsymbol{\mu}}_2, \tilde{\Sigma}_{22}).$$

In the exponent of (3) we have

$$\begin{aligned}
&(\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})^T \tilde{\Sigma}^{-1} (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}}) \\
&= (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1, \tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2)^T \left(\begin{array}{c|c} \tilde{\Sigma}_{11} & 0 \\ \hline 0 & \tilde{\Sigma}_{22} \end{array} \right)^{-1} (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1, \tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2) \\
&= (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1, \tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2)^T \left(\begin{array}{c|c} \tilde{\Sigma}_{11}^{-1} & 0 \\ \hline 0 & \tilde{\Sigma}_{22}^{-1} \end{array} \right) (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1, \tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2) \\
&= (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1)^T \tilde{\Sigma}_{11} (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1) + (\tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2)^T \tilde{\Sigma}_{22} (\tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2).
\end{aligned}$$

Hence, (3) factorizes as follows

$$\begin{aligned}
&f_{\tilde{X}}(\tilde{x}_1, \dots, \tilde{x}_q) \\
&= \frac{1}{\sqrt{(2\pi)^q |\tilde{\Sigma}|}} \exp\Big(-\frac{1}{2} (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})^T \tilde{\Sigma}^{-1} (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}}) \Big) \Big) \\
&= \frac{1}{\sqrt{(2\pi)^{r+s} |\tilde{\Sigma}_{11}| |\tilde{\Sigma}_{22}|}} \exp\Big(-\frac{1}{2} (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1)^T \tilde{\Sigma}_{11} (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1) + (\tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2)^T \tilde{\Sigma}_{22} (\tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2) \Big) \\
&= \frac{1}{\sqrt{(2\pi)^r |\tilde{\Sigma}_{11}|}} \exp\Big(-\frac{1}{2} (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1)^T \tilde{\Sigma}_{11} (\tilde{\mathbf{x}}_1 - \tilde{\boldsymbol{\mu}}_1) \Big) \cdot \frac{1}{\sqrt{(2\pi)^s |\tilde{\Sigma}_{22}|}} \exp\Big(-\frac{1}{2} (\tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2)^T \tilde{\Sigma}_{22} (\tilde{\mathbf{x}}_2 - \tilde{\boldsymbol{\mu}}_2) \Big) \\
&= f_{\tilde{\mathbf{X}}_1}(\tilde{\mathbf{x}}_1) f_{\tilde{\mathbf{X}}_2}(\tilde{\mathbf{x}}_2). \ \blacksquare
\end{aligned}$$