HW5

William Morris

5/5/2020

Main

1

```
load('HW5_BostonHousing.RDAT')
library(knitr)
```

a.

```
#a
Boston$chas<-NULL
```

b.

```
#b
Boston_sample_covariance<-cov(Boston)
kable(data.frame(Boston_sample_covariance))
```

	crim	zn	indus	nox	rm	age	dis	
crim	73.9865782	-40.215956	23.9923388	0.4195939	-1.3250378	85.405322	-6.8767215	46.8477
zn	-40.2159560	543.936814	-85.4126481	-1.3961482	5.1125134	-373.901548	32.6293041	-63.3486
indus	23.9923388	-85.412648	47.0644425	0.6070737	-1.8879566	124.513903	-10.2280975	35.5499
nox	0.4195939	-1.396148	0.6070737	0.0134276	-0.0246034	2.385927	-0.1876958	0.6169
rm	-1.3250378	5.112513	-1.8879566	-0.0246034	0.4936709	-4.751929	0.3036634	-1.2838
age	85.4053223	-373.901548	124.5139031	2.3859272	-4.7519292	792.358399	-44.3293795	111.7708
dis	-6.8767215	32.629304	-10.2280975	-0.1876958	0.3036634	-44.329380	4.4340151	-9.0682
rad	46.8477610	-63.348695	35.5499714	0.6169295	-1.2838146	111.770847	-9.0682520	75.8163
tax	844.8215381	-1236.453735	833.3602902	13.0462855	-34.5834478	2402.690123	-189.6645917	1335.7565
ptratio	5.3993308	-19.776571	5.6921040	0.0473973	-0.5407632	15.936921	-1.0597746	8.7607
race	-302.3818163	373.721402	-223.5797555	-4.0205696	8.2150057	-702.940328	56.0403558	-353.2762
lstat	27.9861679	-68.783037	29.5802703	0.4889462	-3.0797414	121.077725	-7.4733291	30.3854
medv	-30.7185080	77.315175	-30.5208228	-0.4554124	4.4934459	-97.589017	4.8402286	-30.5612

c.

```
#c
Boston_eigen<-eigen(Boston_sample_covariance)
Boston_eigenvalues<-Boston_eigen$values
Boston_eigenvectors<-Boston_eigen$vectors
cat("Eigenvalues: ")
```

```
## Eigenvalues:
```

```
kable(Boston_eigenvalues)
```

x
3.091001e+04
6.250814e+03
8.224633e+02
2.672963e+02
7.704076e+01
4.668341e+01
1.706213e+01
1.355804e+01
8.909243e+00
2.720928e+00
1.101998e+00
2.178827e-01
2.936800e-03

```
cat("Eigenvectors: ")
```

```
## Eigenvectors:
```

```
kable(Boston_eigenvectors)
```

0.0292942	0.0066705	-0.0120191	-0.0253548	-0.2721514	0.9297122	0.1574186	-0.1510899	0.1062494	0.02
-0.0436120	0.0011189	0.6321665	-0.7631562	-0.0919301	-0.0388270	0.0381864	-0.0065478	-0.0572346	0.01
0.0283286	-0.0049445	-0.0883906	0.0130717	-0.0570596	-0.1259603	0.8600071	-0.0784913	-0.4653909	0.00
0.0004497	0.0000019	-0.0018012	-0.0006870	-0.0001476	-0.0004479	0.0048466	0.0011563	-0.0032793	-0.01
-0.0011707	0.0003609	0.0050256	-0.0063810	0.0487361	0.0175790	-0.0120371	-0.0106846	-0.0037834	-0.00
0.0836334	-0.0056920	-0.7524508	-0.6406005	0.0757549	-0.0043150	-0.0783141	-0.0615696	-0.0074110	0.00
-0.0065593	0.0003554	0.0447027	-0.0017231	-0.0351605	-0.0250886	-0.1104212	-0.0409772	0.0122137	0.11
0.0449866	-0.0086373	0.0032929	0.0185548	0.0399569	0.2307916	-0.3598028	0.3997429	-0.7970284	-0.13
0.9494105	-0.2926719	0.0952212	0.0191954	0.0246645	-0.0260119	-0.0030239	-0.0156041	0.0445704	0.00
0.0056030	-0.0025220	-0.0116074	0.0329309	-0.0633674	-0.0127482	-0.0879648	-0.0442730	-0.1535177	0.97
-0.2911764	-0.9560583	-0.0247914	-0.0031965	-0.0203590	0.0103770	0.0004885	0.0024118	-0.0028663	-0.00
0.0229612	0.0058043	-0.0948194	-0.0399403	-0.4601069	-0.0681612	0.1693334	0.8127394	0.2766421	0.06
-0.0255283	-0.0088437	0.0733563	-0.0544337	0.8285790	0.2417580	0.2231047	0.3779331	0.1768331	0.13

```
remove(Boston_eigen)
```

d.

From below, you can see the difference between S and $V\Lambda V^T$. So, they are essentially equal, allowing for some float-level inaccuracies.

```
#d
Lambda<-diag(Boston_eigenvalues)
v_lambda_v<-Boston_eigenvectors%*%Lambda%*%t(Boston_eigenvectors)
kable(data.frame(Boston_sample_covariance - v_lambda_v))
```

	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	race	lstat	medv
crim	0	0	0	0	0	0	0	0	0	0	0	0	0
zn	0	0	0	0	0	0	0	0	0	0	0	0	0
indus	0	0	0	0	0	0	0	0	0	0	0	0	0
nox	0	0	0	0	0	0	0	0	0	0	0	0	0

	crim	zn	indus	nox	rm	age	dis	rad	tax	ptratio	race	lstat	medv
rm	0	0	0	0	0	0	0	0	0	0	0	0	0
age	0	0	0	0	0	0	0	0	0	0	0	0	0
dis	0	0	0	0	0	0	0	0	0	0	0	0	0
rad	0	0	0	0	0	0	0	0	0	0	0	0	0
tax	0	0	0	0	0	0	0	0	0	0	0	0	0
ptratio	0	0	0	0	0	0	0	0	0	0	0	0	0
race	0	0	0	0	0	0	0	0	0	0	0	0	0
lstat	0	0	0	0	0	0	0	0	0	0	0	0	0
medv	0	0	0	0	0	0	0	0	0	0	0	0	0

```
remove(Lambda,v_lambda_v)
```

e.

```
#e
pca<-princomp(Boston)
print(summary(pca))
```

```
## Importance of components:
##                             Comp.1     Comp.2      Comp.3     Comp.4
## Standard deviation     175.6386229 78.9839276 28.65026817 16.3330343
## Proportion of Variance   0.8045735  0.1627058  0.02140834  0.0069576
## Cumulative Proportion    0.8045735  0.9672794  0.98868773  0.9956453
##                             Comp.5      Comp.6       Comp.7       Comp.8
## Standard deviation     8.768608846 6.825771401 4.1265499050 3.6784846027
## Proportion of Variance 0.002005336 0.001215148 0.0004441196 0.0003529097
## Cumulative Proportion  0.997650662 0.998865810 0.9993099299 0.9996628396
##                              Comp.9      Comp.10      Comp.11      Comp.12
## Standard deviation     2.9818846230 1.6478926312 1.048723e+00 4.663176e-01
## Proportion of Variance 0.0002319035 0.0000708245 2.868452e-05 5.671388e-06
## Cumulative Proportion  0.9998947432 0.9999655677 9.999943e-01 9.999999e-01
##                             Comp.13
## Standard deviation     5.413879e-02
## Proportion of Variance 7.644390e-08
## Cumulative Proportion  1.000000e+00
```

f.

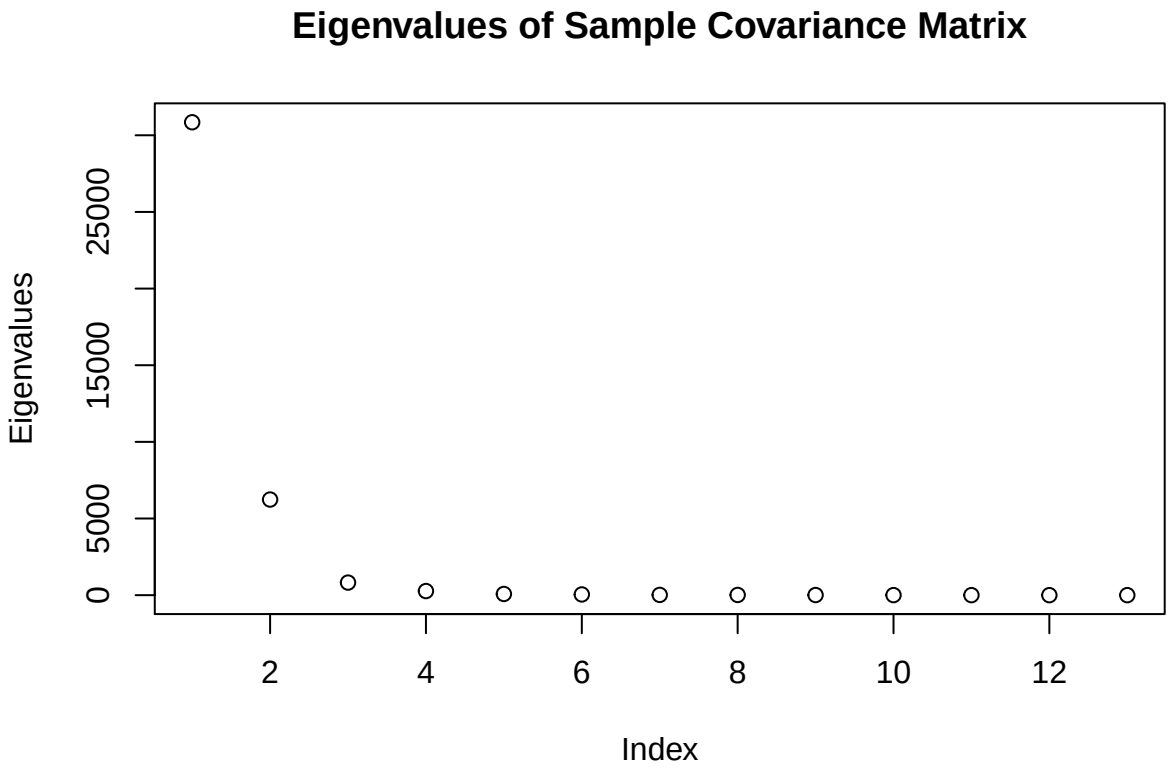
```
#f
Boston_pca_variance<-(pca$sdev)^2
kable(data.frame(Boston_pca_variance))
```

	Boston_pca_variance
Comp.1	3.084893e+04
Comp.2	6.238461e+03
Comp.3	8.208379e+02
Comp.4	2.667680e+02
Comp.5	7.688850e+01
Comp.6	4.659116e+01
Comp.7	1.702841e+01
Comp.8	1.353125e+01

	Boston_pca_variance
Comp.9	8.891636e+00
Comp.10	2.715550e+00
Comp.11	1.099821e+00
Comp.12	2.174521e-01
Comp.13	2.931000e-03

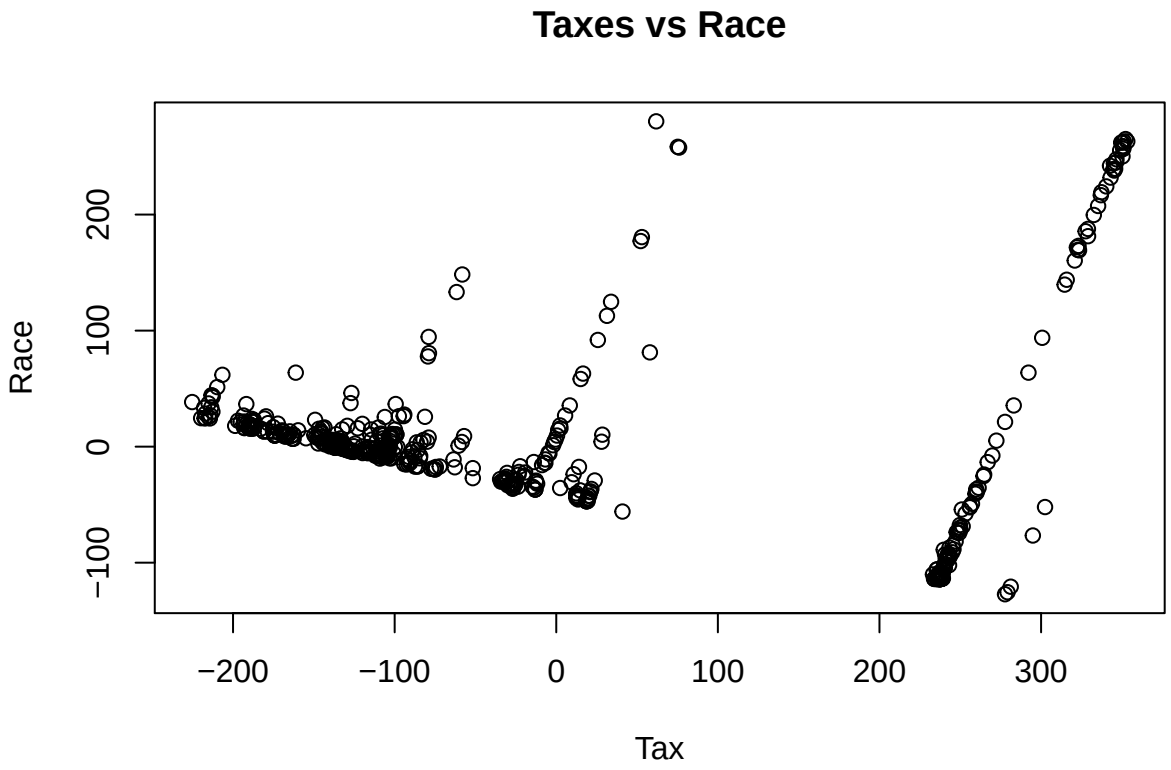
g.

```
#g
plot(Boston_pca_variance,ylab='Eigenvalues',main='Eigenvalues of Sample Covariance Matrix')
```



h.

```
#h
plot(pca$scores[,1],pca$scores[,2],xlab='Tax',ylab='Race',main='Taxes vs Race')
```



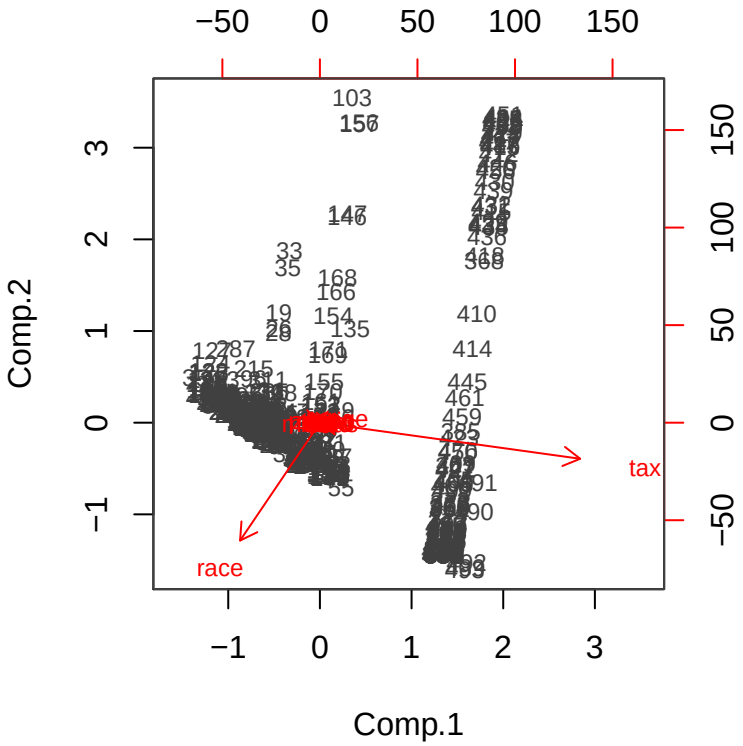
I don't know which version is more current, but the documentation for the Boston housing data on R Studio doesn't have a "race" component. Instead, it has a variable called "black" which attepmts to give the average number of black people in a house's neighborhood. That being said, there is definitely a negative correlation between taxes and the race of a neighborhood, with a distinct separation into 2 clusters, implying white and black neighborhoods.

Also, I count 4 places where the data shows a significant positive correlation withing the larger graph. From this, I gather that some neighborhoods have become gentrified while keeping racial diversity. Although, this is not the majority.

i.

```
#i
biplot(pca,choices=c(1,2),col=c('gray25','red'),cex=.75,pc.biplot = T)
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length
## = arrow.len): zero-length arrow is of indeterminate angle and so skipped
```



Taxes and race are the two most significant components. Everything else is jumbled up near the center, implying their variances are quite small. From the data below, we can see that the first two components account for 96% of the variance.

```
pca.V<-pca$loadings
pca.L <- pca$sdev^2
{
  p1 <- sapply(1:13,function(u) pca.L[u]/sum(pca.L))
  p2 <- sapply(1:13,function(u) sum(pca.L[1:u])/sum(pca.L))
  data.frame(Proportion_of_Total_Variance=p1,Percentage_Explained=p2)
}
```

```
##         Proportion_of_Total_Variance Percentage_Explained
## Comp.1                  8.045735e-01            0.8045735
## Comp.2                  1.627058e-01            0.9672794
## Comp.3                  2.140834e-02            0.9886877
## Comp.4                  6.957600e-03            0.9956453
## Comp.5                  2.005336e-03            0.9976507
## Comp.6                  1.215148e-03            0.9988658
## Comp.7                  4.441196e-04            0.9993099
## Comp.8                  3.529097e-04            0.9996628
## Comp.9                  2.319035e-04            0.9998947
## Comp.10                 7.082450e-05            0.9999656
## Comp.11                 2.868452e-05            0.9999943
## Comp.12                 5.671388e-06            0.9999999
## Comp.13                 7.644390e-08            1.0000000
```

2

```
library("readxl")
```

```
## Warning: package 'readxl' was built under R version 3.5.2
```

```
hw5_wiki<-read_excel('HW5_wiki.xlsx')
cat("Dimensions: ", dim(hw5_wiki))
```

```
## Dimensions:  913 26
```

a

```
hw5<-hw5_wiki[complete.cases(hw5_wiki),]
cat("Dimensions: ", dim(hw5))
```

```
## Dimensions:  703 26
```

b

```
f <- factanal(hw5,factors=6)
```

c

Below are the loadings for the factor analysis. The 6 factors identified are common to all variables in the data. Each entry in the matrix shows the percent of that variable which is explained by a particular factor. I've muted all loadings which are less than 50% so that we can see the dominant factor for each variable. As you can see, some variables have empty rows, which means no factor accounted for more than 50% of its variance. That means these variables are unique from the data as a whole and can't be explained by the same common factors.

```
print(f$loadings, digits=3, cutoff=0.5, sort=FALSE)
```

```
##
## Loadings:
##     Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
## Q1                                           0.603
## Q2                                           0.546
## Q3   0.525
## Q4                   0.561
## Q5                   0.564
## Q6
## Q7                   0.555
## Q8                   0.685
## Q9           0.820
## Q10          0.735
## Q11          0.651
## Q12
## Q13          0.573
## Q14                          0.752
## Q15                          0.730
## Q16                          0.738
## Q17  0.627
## Q18  0.579
## Q19  0.884
## Q20  0.850
```

```
## Q21  0.508
## Q22  0.595
## Q23                                  0.660
## Q24
## Q25
## Q26
##
##                Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
## SS loadings       4.43   2.982   2.423   2.061   1.137   1.099
## Proportion Var    0.17   0.115   0.093   0.079   0.044   0.042
## Cumulative Var    0.17   0.285   0.378   0.458   0.501   0.544
```

d

Below is the list of each variable's uniqueness, sorted in ascending order. If you check the most unique variables at the bottom, you'll see most of them are the same variables which had no dominant factor in part (c).

Uniqueness measures the opposite of the loadings. That is, how much of a variable's variance is due to unique factors which the other variables don't have. The further down this list you go, the less likely it is that a variable shares anything in common with another variable.

```
kable(data.frame(sort(f$uniquenesses)))
```

	sort.f.uniquenesses.
Q19	0.1023162
Q20	0.1724038
Q9	0.2035531
Q23	0.2090715
Q1	0.2710815
Q22	0.2855465
Q2	0.3096658
Q3	0.3135616
Q14	0.3353596
Q10	0.3417101
Q16	0.3580430
Q15	0.4013338
Q21	0.4158639
Q8	0.4438418
Q13	0.4746038
Q11	0.4754431
Q17	0.4987854
Q24	0.5034588
Q7	0.5299136
Q5	0.6011718
Q4	0.6166603
Q18	0.6294818
Q25	0.8074190
Q6	0.8292927
Q26	0.8445037
Q12	0.8937166

e

The code below is a function I found for organizing variabes according to their dominant factor and displaying it neatly.

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 3.5.2
```

```
library(GPArotation)
f_psych <- fa(hw5,nfactors=6,covar=T,fm='pa')
factor2cluster(f_psych,aslist = TRUE)
```

```
## $PA1
## [1] "Q1"  "Q2"  "Q3"  "Q7"  "Q19" "Q21"
##
## $PA5
## [1] "Q22" "Q23" "Q24"
##
## $PA3
## [1] "Q17" "Q18" "Q25" "Q26"
##
## $PA6
## [1] "Q9"  "Q10" "Q11" "Q13"
##
## $PA2
## [1] "Q12" "Q14" "Q15" "Q16"
##
## $PA4
## [1] "Q4"   "Q5"   "Q6"   "Q8"   "-Q20"
```

f

Factors 1 and 4 are the most significant. Factor 5 is the least significant. This can be seen in how many variables had them as their dominant factor above.

3

```
banknotes<-read.csv('HW5_banknotes.csv')
banknotes$X<-NULL
```

a

```
km<-kmeans(banknotes, centers = 2)
```

b

Cluster 1

```
rownames(banknotes[which(km$cluster==1),])
```

```
##  [1] "2"  "5"  "6"  "8"  "10" "13" "16" "20" "21" "22" "23" "24" "25" "27"
## [15] "28" "30" "31" "32" "38" "39" "41" "42" "44" "45" "48" "51" "56" "58"
## [29] "59" "60" "62" "66" "67" "70" "72" "73" "75" "76" "78" "80" "82" "83"
## [43] "85" "89" "90" "95" "98" "99"
```

Cluster 2

```
rownames(banknotes[which(km$cluster==2),])
```

```
##  [1] "1"   "3"   "4"   "7"   "9"   "11"  "12"  "14"  "15"  "17"  "18"
## [12] "19"  "26"  "29"  "33"  "34"  "35"  "36"  "37"  "40"  "43"  "46"
## [23] "47"  "49"  "50"  "52"  "53"  "54"  "55"  "57"  "61"  "63"  "64"
## [34] "65"  "68"  "69"  "71"  "74"  "77"  "79"  "81"  "84"  "86"  "87"
## [45] "88"  "91"  "92"  "93"  "94"  "96"  "97"  "100"
```

c

Cluster for the 11th Banknote

```
print(km$cluster[11])
```

```
## [1] 2
```

d

Counterfeit Banknotes

```
genuine<-km$cluster[51]
counterfeit<-rownames(banknotes[which(km$cluster != genuine),])
print(counterfeit)
```

```
##  [1] "1"   "3"   "4"   "7"   "9"   "11"  "12"  "14"  "15"  "17"  "18"
## [12] "19"  "26"  "29"  "33"  "34"  "35"  "36"  "37"  "40"  "43"  "46"
## [23] "47"  "49"  "50"  "52"  "53"  "54"  "55"  "57"  "61"  "63"  "64"
## [34] "65"  "68"  "69"  "71"  "74"  "77"  "79"  "81"  "84"  "86"  "87"
## [45] "88"  "91"  "92"  "93"  "94"  "96"  "97"  "100"
```

Graduate

1

a

In general $S = \frac{1}{n-1}X^TX$. Also, by the Spectral Decomposition Theorem $S = V\Lambda V^T$ with V being the matrix formed by the eigenvectors of S and $\Lambda = \text{diag}\{\lambda_1, \ldots, \lambda_p\}$ for the eigenvalues of S. Lastly, recall that $V^TV = I_p$. Then,

$$\begin{aligned}
S_Y &= \frac{1}{n-1}Y^TY \\
&= \frac{1}{n-1}(XV)^T(XV), \text{ by the definition of Pricipal Components} \\
&= \frac{1}{n-1}V^T(X^TX)V \\
&= \frac{1}{n-1}(n-1)V^TS_XV \\
&= V^T(V\Lambda V^T)V \\
&= (V^TV)\Lambda(V^TV) \\
&= \Lambda
\end{aligned}$$

b

$$\begin{aligned}
trace(S_X) &= trace(VV^TS_X) \\
&= trace(V^TS_XV) \\
&= trace(\Lambda) \\
&= trace(S_Y)
\end{aligned}$$

2

Consider that Y is a linear transformation of X. Then, for $Y = AX + b$, $\Sigma_Y = A\Sigma_XA^T$. Let $A = U^T$ and $b^T = [0 \ldots 0]$.

$$\begin{aligned}
\Sigma_Y &= U^T\Sigma_XU^{TT} \\
&= U^TUEU^TU \\
&= E \\
&= diag\{e_1, \ldots, e_p\}
\end{aligned}$$

a

$$Var(Y_j) = \sigma_{jj} \in \Sigma_Y = e_j, \ \forall j$$

b

$$Cov(Y_i, Y_j) = \sigma_{ij} \in \Sigma_Y \ s.t. \ i \neq j = 0.$$

3

Specifically, we need to assume $Var[F] = I,\ Var[U] = \Psi,\ Cov(F, U) = 0$. Then,

$$\begin{aligned}
\Sigma_X = Var[X] &= Var[QF + U] \\
&= QVar[F]Q^T + Var[U] + 2Cov(F, U) \\
&= QIQ^T + \Psi + 2 \cdot 0 \\
&= QQ^T + \Psi
\end{aligned}$$

4

$$\begin{aligned}
l(X, \bar{x}, \Sigma) &= logL(X, \bar{x}, \Sigma) \\
&= \sum_{i=1}^{n} log\left[det(2\pi\Sigma)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}(x - \bar{x})^T\Sigma^{-1}(x - \bar{x})\right\}\right] \\
&= \sum_{i=1}^{n} log\left[det(2\pi\Sigma)^{-\frac{1}{2}}\right] + log\left[exp\left\{-\frac{1}{2}(x - \bar{x})^T\Sigma^{-1}(x - \bar{x})\right\}\right] \\
&= -\frac{1}{2}nlog[det(2\pi\Sigma)] + \sum_{i=1}^{n}\left[-\frac{1}{2}(x - \bar{x}^T)\Sigma^{-1}(x - \bar{x})\right] \\
&= -\frac{1}{2}\left\{nlog[det(2\pi\Sigma)] + \sum_{i=1}^{n}[(x - \bar{x})^T\Sigma^{-1}(x - \bar{x})]\right\} \\
&= -\frac{1}{2}\left\{nlog[det(2\pi\Sigma)] + tr[(x - \bar{x})^T\Sigma^{-1}(x - \bar{x})]\right\} \\
&= -\frac{1}{2}\left\{nlog[det(2\pi\Sigma)] + tr[\Sigma^{-1}(x - \bar{x})(x - \bar{x})^T]\right\} \\
&= -\frac{1}{2}\left\{nlog[det(2\pi\Sigma)] + tr[\Sigma^{-1}(n - 1)S]\right\} \\
&= -\frac{1}{2}\left\{nlog[det(2\pi\Sigma)] + (n - 1)tr[\Sigma^{-1}S]\right\}
\end{aligned}$$