INF 511 Assignment 1

Jen Diehl (6179236), Sam Watson (6174574), Natasha Wesely (6180693)

2022-09-06

Question 1

```
set.seed(24601)
(myvec<- round(rnorm(n=30), digits=1))
```

```
##  [1] -0.3  0.6  0.5 -1.9 -0.3 -0.6  0.3 -0.3 -1.2 -3.2 -0.6 -0.3 -1.0 -0.2  1.3
## [16] -0.4  0.0 -0.7  0.0  0.7  1.0 -0.1 -1.0  0.6  0.5  0.8 -0.7  0.7  0.2  0.4
```

```
#(a)
sum(myvec)
```

```
## [1] -5.2
```

```
#(b)
prod(myvec)
```

```
## [1] 0
```

```
#(c)
myvec^2
```

```
##  [1]  0.09  0.36  0.25  3.61  0.09  0.36  0.09  0.09  1.44 10.24  0.36  0.09
## [13]  1.00  0.04  1.69  0.16  0.00  0.49  0.00  0.49  1.00  0.01  1.00  0.36
## [25]  0.25  0.64  0.49  0.49  0.04  0.16
```

```
#(d)
exp(-myvec)
```

exp(myvec) -1

```
##  [1]  1.3498588  0.5488116  0.6065307  6.6858944  1.3498588  1.8221188
##  [7]  0.7408182  1.3498588  3.3201169 24.5325302  1.8221188  1.3498588
## [13]  2.7182818  1.2214028  0.2725318  1.4918247  1.0000000  2.0137527
## [19]  1.0000000  0.4965853  0.3678794  1.1051709  2.7182818  0.5488116
## [25]  0.6065307  0.4493290  2.0137527  0.4965853  0.8187308  0.6703200
```

```
#(e)
log(myvec)
```

```
##  [1]        NaN -0.5108256 -0.6931472        NaN        NaN        NaN
##  [7] -1.2039728        NaN        NaN        NaN        NaN        NaN
## [13]        NaN        NaN  0.2623643        NaN       -Inf        NaN
## [19]       -Inf -0.3566749  0.0000000        NaN        NaN -0.5108256
## [25] -0.6931472 -0.2231436        NaN -0.3566749 -1.6094379 -0.9162907
```

```
# Problem is can't compute the natural log of a negative element
#(f)
log10(myvec)
```

```
##  [1]         NaN -0.22184875 -0.30103000         NaN         NaN         NaN
##  [7] -0.52287875         NaN         NaN         NaN         NaN         NaN
## [13]         NaN         NaN  0.11394335         NaN        -Inf         NaN
## [19]        -Inf -0.15490196  0.00000000         NaN         NaN -0.22184875
## [25] -0.30103000 -0.09691001         NaN -0.15490196 -0.69897000 -0.39794001
```

```
#(g) Compute the average, variance, standard deviation, median and the three quartiles of the of the elements in
the vector. (2 points)
mean(myvec)
```

```
## [1] -0.1733333
```

```
var(myvec)
```

```
## [1] 0.844092
```

```
sd(myvec)
```

```
## [1] 0.9187448
```

```
median(myvec)
```

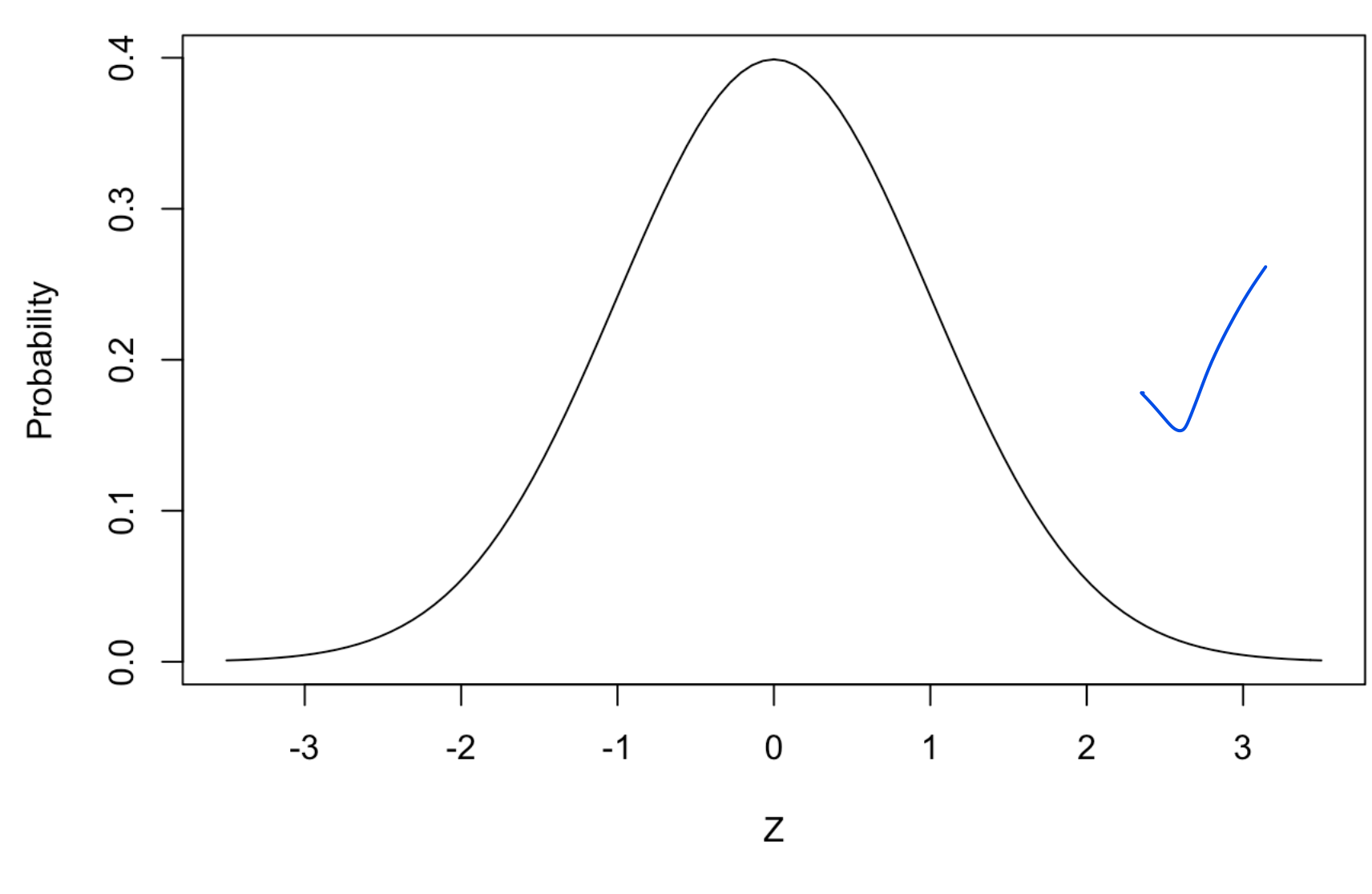
```
## [1] -0.15
```

```
quantile(myvec, probs=c(.25,.50,.75))
```

```
##   25%   50%   75%
## -0.60 -0.15  0.50
```

Question 2

```
curve(expr = dnorm, from = -3.5, to = 3.5, ylab = "Probability", xlab = "Z", main = "Standard Normal Distribution
")
```



Question 3

```
theta = 2.576
pnorm(theta)
```

```
## [1] 0.9950025
```

Question 4

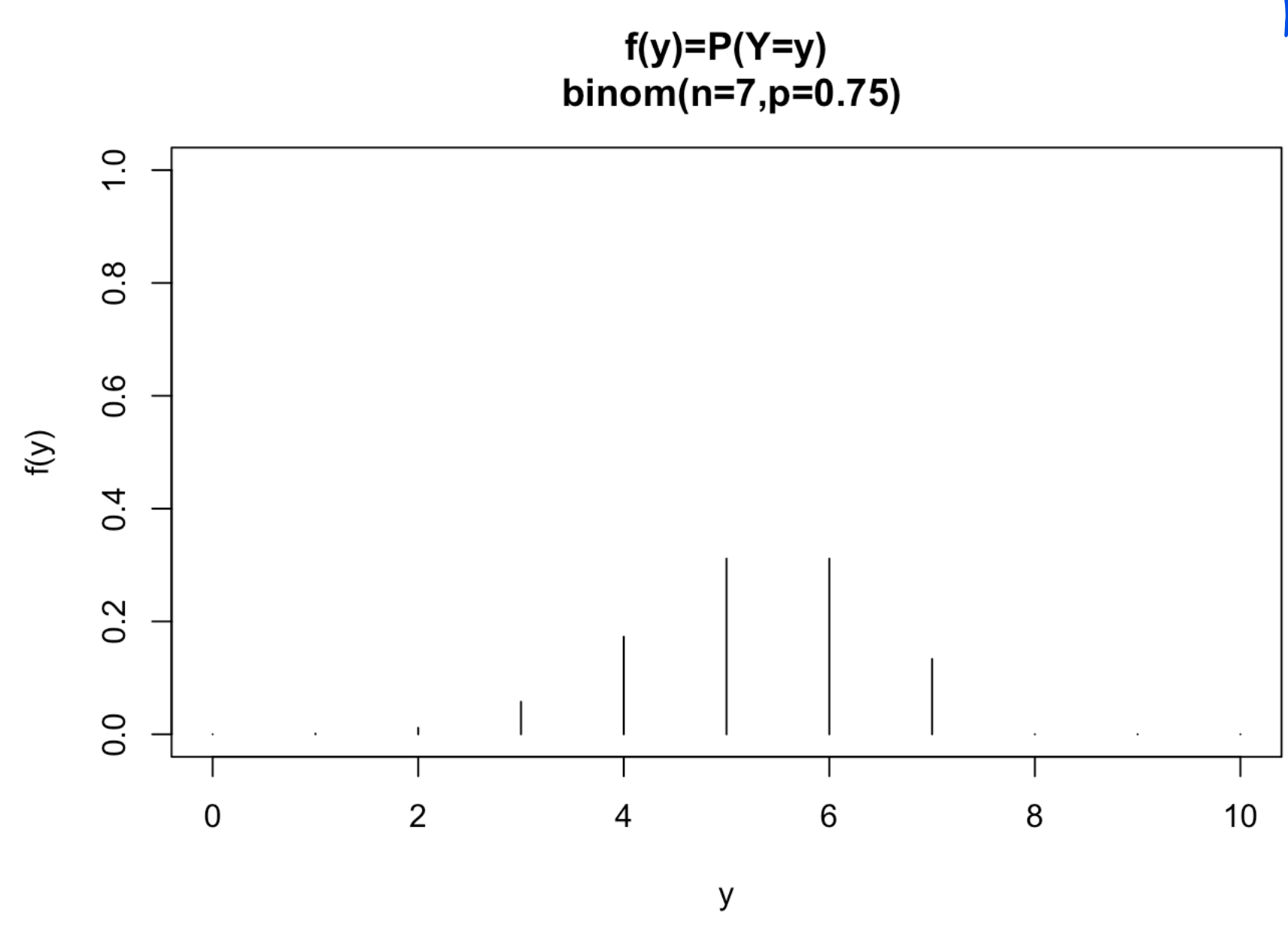
```
qnorm(0.995)
```

```
## [1] 2.575829
```

Question 5

```
plot(0:10, binprobs<- dbinom(0:10,size=7,p=0.75),
ylab="f(y)", xlab="y", type="h",
pch=20, main="f(y)=P(Y=y)\n binom(n=7,p=0.75)", ylim=c(0,1))
```

make sure to adjust the x and y lim



Question 6

```
pbinom(q = 2 ,size = 7, prob = 0.75, lower.tail = F) + dbinom(2,size = 7, prob = 0.75)
```

```
## [1] 0.9986572
```

Question 7

```
pbinom(q =1.5 ,size = 7, prob = 0.75, lower.tail = F) + dbinom(1.5,size = 7, prob = 0.75)
```

```
## [1] 0.9986572
```

Question 8

Mean: $5 + \mu_1 + 2\mu_2$ Covariance: 0 because they are independent variables. Variance: $\sigma_1^2 + 4\sigma_2^2$ It is normally distributed.

Question 9

```
set.seed(5551212)
(A<- matrix(round(rnorm(n=8,sd=3)),nrow=2,ncol=4, byrow=TRUE))
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    2   -4   -5
## [2,]    4    5    1    3
```

```
(B<- matrix(round(rnorm(n=4,sd=3)),nrow=1,ncol=4))
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    4    1    5   -5
```

```
(C<- matrix(round(rnorm(n=8,sd=3)),nrow=2,ncol=4, byrow=TRUE))
```

```
##      [,1] [,2] [,3] [,4]
## [1,]   -1    2    2    3
## [2,]   -3    4    4    0
```

```
#a)
A[2,2]
```

```
## [1] 5
```

```
#b)
matrix(A[,2])
```

```
##      [,1]
## [1,]    2
## [2,]    5
```

```
#c)
(D <- A +C); dim(D)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    0    4   -2   -2
## [2,]    1    9    5    3
```

```
## [1] 2 4
```

```
#d)
(E <- A - C); dim(E)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    2    0   -6   -8
## [2,]    7    1   -3    3
```

```
## [1] 2 4
```

```
#e)
(F<-A%*%t(B)); dim(F)
```

```
##      [,1]
## [1,]   11
## [2,]   11
```

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

```
#f)
(G<-A%*%t(C)); dim(G)
```

```
##      [,1] [,2]
## [1,]  -20  -11
## [2,]   17   12
```

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

```
#g)
(H <- t(C)%*%A); dim(H)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  -13  -17    1   -4
## [2,]   18   24   -4    2
## [3,]   18   24   -4    2
## [4,]    3    6  -12  -15
```

```
## [1] 4 4
```

h. A%%C *Error in A* %% C : non-conformable arguments It doesn't work because the columns of the first matrix and rows of the second matrix must be the same size.

X (A*C) -2

Question 10

```
set.seed(sum(c(90620,5150)))
X<- matrix(c(rep(1,6),round(runif(n=6,min=0,max=10))), ncol=2)
Y <- 4.5 + 1.5*X[,2,drop=FALSE] + round(rnorm(6),2)
ord<- order(Y)
(X<- X[ord,])
```

```
##      [,1] [,2]
## [1,]    1    2
## [2,]    1    3
## [3,]    1    6
## [4,]    1    6
## [5,]    1    6
## [6,]    1    8
```

```
(Y<- Y[ord,,drop=FALSE])
```

```
##       [,1]
## [1,]  6.74
## [2,]  8.59
## [3,] 12.94
## [4,] 12.95
## [5,] 13.95
## [6,] 17.84
```

```
#a)
(A<-t(X)%*%Y)
```

```
##        [,1]
## [1,]  73.01
## [2,] 421.01
```

```
#b)
(B <- t(X)%*%X)
```

```
##      [,1] [,2]
## [1,]    6   31
## [2,]   31  185
```

```
#c)
(Binv <- solve(B))
```

```
##            [,1]        [,2]
## [1,]  1.2416107 -0.20805369
## [2,] -0.2080537  0.04026846
```

```
#d)
round(Binv%*%B,2)
```

```
##      [,1] [,2]
## [1,]    1    0
## [2,]    0    1
```

```
#e)
Binv%*%A
```

```
##          [,1]
## [1,] 3.057315
## [2,] 1.763423
```

```
#f)
#Answer to problem D is symmetric because it is the identity matrix.

# g)
model = lm(Y ~ X[,2])
summary(model)
```

```
##
## Call:
## lm(formula = Y ~ X[, 2])
##
## Residuals:
##       1       2       3       4       5       6
##  0.1558  0.2424 -0.6979 -0.6879  0.3121  0.6753
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0573     0.7040   4.343 0.012226 *
## X[, 2]        1.7634     0.1268  13.909 0.000155 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6318 on 4 degrees of freedom
## Multiple R-squared:  0.9797, Adjusted R-squared:  0.9747
## F-statistic: 193.5 on 1 and 4 DF,  p-value: 0.0001549
```

```
print("The intercept (B0) = 3.057 and the slope (B1) = 1.763")
```

```
## [1] "The intercept (B0) = 3.057 and the slope (B1) = 1.763"
```

```
# h)
# Var(B) = (XtX)-1XtVar(y)X(XtX)-1
varY = 0.3992
varB = Binv%*%t(X)%*%X%*%Binv*varY
sqrt(diag(varB))
```

```
## [1] 0.7040249 0.1267879
```

```
summary(model)$coefficients
```

```
##             Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 3.057315  0.7039938  4.342816 0.0122258563
## X[, 2]      1.763423  0.1267823 13.909063 0.0001549317
```

```
print("The parameter variances computed by hand are also identical to the parameter standard errors from the line
ar model (created with lm()).")
```

```
## [1] "The parameter variances computed by hand are also identical to the parameter standard errors from the lin
ear model (created with lm())."
```