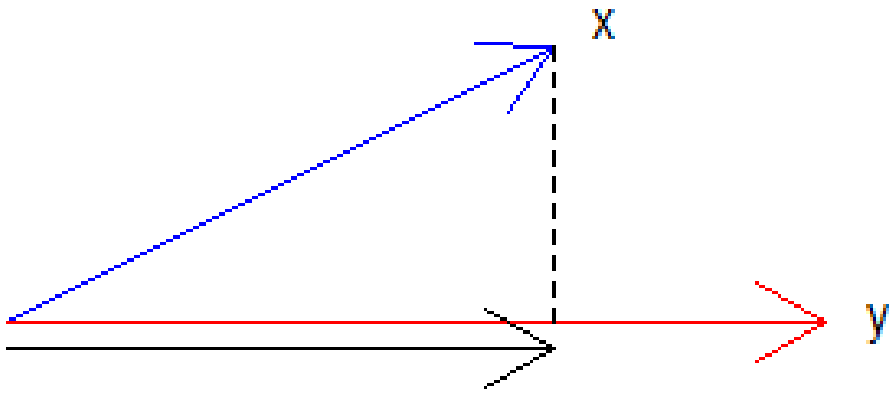
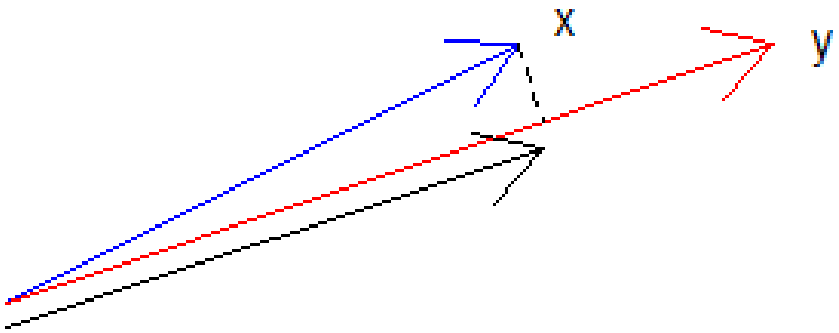
```
###Exercise 1

x1 = c(1,4,1,3,2,-6)
x2 = c(-1,5,2,4,1,-1)
x3 = c(-1,5,1,0,-1,3)

### a)
2*x1-x2+x3

## [1]  2  8  1  2  2 -8

### b)
###  i)
sum(x1*x2)

## [1] 41

###  ii)
sum(x2*x3)

## [1] 24

###  iii)
sum(x1*x3)

## [1] 0

#x1 und x3 sind orthogonal zu einander"

### c)
x1%*%t(x2)

##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]   -1    5    2    4    1   -1
## [2,]   -4   20    8   16    4   -4
## [3,]   -1    5    2    4    1   -1
## [4,]   -3   15    6   12    3   -3
## [5,]   -2   10    4    8    2   -2
## [6,]    6  -30  -12  -24   -6    6

### d)
L1 = norm(as.matrix(x1),type="2")
L1

## [1] 8.185353

L2 = norm(as.matrix(x2),type="2")
L2

## [1] 6.928203

L3 = norm(as.matrix(x3),type="2")
L3

## [1] 6.082763

### d) alternativ
L1 = sqrt(sum(x1^2))
L1
```

```
## [1] 8.185353

L2 = sqrt(sum(x2^2))
L2

## [1] 6.928203

L3 = sqrt(sum(x3^2))
L3

## [1] 6.082763

### e) Calculate the projection of
###  i) x1 on x2
pro12 = (((t(x2)%*%x1)/L2)%*%(1/L2))%*%x2
pro12

##            [,1]     [,2]     [,3]     [,4]      [,5]       [,6]
## [1,] -0.8541667 4.270833 1.708333 3.416667 0.8541667 -0.8541667

### ii) x1 on x3
pro13 = (((t(x3)%*%x1)/L3)%*%(1/L3))%*%x3
pro13

##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0    0    0    0    0    0

#Mit diesem Ergebnis war zu rechnen, da diese beiden Vektoren orthogonal zu
einander sind.
```

```
###Exercise 3

remove(list = ls())

### a)
x = 0
y = 0
x1 = c(2,1)
y1 = c(3,0)

lx1 = sqrt(sum(x1^2))
ly1 = sqrt(sum(y1^2))
projxy = (((t(y1)%*%x1)/ly1)%*%(1/ly1))%*%y1
projxy

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

plot(x,y,xlim=c(0,4),ylim=c(-0.2,1.2),type="n",axes=FALSE,frame.plot=FALSE,
ann=FALSE,asp=1)
arrows(0,0,2,1,col = 4)
arrows(0,0,3,0,col = 2)
arrows(0,-0.1,2,-0.1)
arrows(2,1,projxy[1,1],projxy[1,2], lty = 2, code = 0)
arrows(2,1,2,0, lty = 2, code = 0)
text(2.2,1.1,"x")
text(3.2,0,"y")
```



```
### b)
x = 0
y = 0
x1 = c(2,1)
y1 = c(3,1)

lx1 = sqrt(sum(x1^2))
ly1 = sqrt(sum(y1^2))
projxy = (((t(y1)%*%x1)/ly1)%*%(1/ly1))%*%y1
projxy

##      [,1] [,2]
## [1,]  2.1  0.7

plot(x,y,xlim=c(0,4),ylim=c(-0.2,1.2),type="n",axes=FALSE,frame.plot=FALSE,
ann=FALSE,asp=1)
arrows(0,0,2,1,col = 4)
arrows(0,0,3,1,col = 2)
arrows(0,-0.1,projxy[1,1],projxy[1,2] -0.1)
arrows(2,1,projxy[1,1],projxy[1,2], lty = 2, code = 0)
text(2.2,1.1,"x")
text(3.2,1,"y")
```




```
### c)
?plot

## starting httpd help server ... done

#Mit asp wird das Längenverhältnis zwischen den Einheiten der y-Achse und
der x-Achse (y/x) bestimmt. asp = 1 garantiert, dass das Verhältnis, zwisch
en den Längeneinheiten 1 beträgt.
```

```
###Exercise 4

remove(list = ls())

mv1 = c(3,2,1,0,2,1,-1,4,3)
m1 = matrix(mv1,3,3)
mv2 = c(1,2,-1,5,1,3)
m2 = matrix(mv2,2,3, byrow = TRUE)
v1 = c(1,4,1)
v2 = c(-1,5,2)

###  i)
m1%*%v1

##      [,1]
## [1,]    2
## [2,]   14
## [3,]    8

#Diese Matrix ist nicht symmetrisch

###  ii)
m2%*%v1

##      [,1]
## [1,]    8
## [2,]   12

#Diese Matrix ist nicht symmetrisch

###  iii)
t(v1)%*%m1%*%v1

##      [,1]
## [1,]   66

#Diese Matrix ist nicht symmetrisch

###  iv)
m2%*%m1

##      [,1] [,2] [,3]
## [1,]    6    3    4
## [2,]   20    5    8

#Diese Matrix ist nicht symmetrisch

###  v)
t(m1)%*%m1

##      [,1] [,2] [,3]
## [1,]   14    5    8
## [2,]    5    5   11
## [3,]    8   11   26

#Diese Matrix ist symmetrisch
```

```
###  vi)
t(m2)%*%m2
```

```
##      [,1] [,2] [,3]
## [1,]   26    7   14
## [2,]    7    5    1
## [3,]   14    1   10
```

```
#Diese Matrix ist symmetrische
```

```
###  vii)
m1%*%t(m1)
```

```
##      [,1] [,2] [,3]
## [1,]   10    2    0
## [2,]    2   24   16
## [3,]    0   16   11
```

```
#Diese Matrix ist symmetrisch
```

```
###  viii)
m2%*%t(m2)
```

```
##      [,1] [,2]
## [1,]    6    4
## [2,]    4   35
```

```
#Diese Matrix ist symmetrisch
```

```
###Exercise 5

remove(list = ls())

av = c(2,1,3,4,3,8,-2,2,0,-4,5,1,-1,3,4,1)
A = matrix(av,4,4)

### a)
det(A)

## [1] 238

#Die Determinante(A) ist !=0. Daraus ist zu folgern, dass die Matrix vollen
Rang hat und die Spaltenvektoren linear unabhängig sind.

### b)
mv2 = c(1,2,-1,5,1,3)
m2 = matrix(mv2,2,3, byrow = TRUE)

dm2 = det(t(m2)%*%m2)
round(dm2)

## [1] 0

#Die Determinante(t(m2)%*%m2)) = 0. Somit hat die Matrix keinen vollen Rang
und linear abhängige Spaltenvektoren. Es existiert keine Inverse zu dieser
Matrix

### c)
iA = solve(A)
round(iA, digits = 4)

##         [,1]    [,2]    [,3]    [,4]
## [1,] -0.2857 -0.0630 -0.1555  0.5252
## [2,]  0.4286  0.0798  0.1303 -0.3319
## [3,]  0.5714 -0.0210  0.2815 -0.4916
## [4,] -0.2857  0.1134  0.0798  0.0546
```

```
###Exercise 6

remove(list = ls())

vx = c(1,0,0,0,0,1)
x = matrix(vx,3,2)
y = c(1,3,2)

yd = x%*%(solve(t(x)%*%x))%*%t(x)%*%y
yd

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

```
###Exercise 7
```

## Declaration of Honor

I hereby declare that I will respect all rules mentioned above. Specifically, I will neither copy exercise solutions from other students or other sources nor will I let copy other students from my own solutions. I accept being excluded from this course or fined with a deduction of exercise points if I offend against these rules.

| Student Name: | Date: | Signature: |
|---|---|---|
| Benjamin C. Herbert | 16.09.2019 | |