Question 1

Recall from Section 8.3.8 in Prof. Veraart's notes that the p.d.f. of the standard Cauchy distribution is

$$f_X(x) = \frac{1}{\pi\left(1+x^2\right)}, \qquad \text{with support } x \in \mathbb{R}.$$

(a) Show that f_X is a probability density function (p.d.f.) and plot $y = f_X(x)$.

(b) Compute the first (raw) moment of X, $\mu = \mu_1' = \mathrm{E}(X)$.

(c) Compute the kth central moment of X, $\mu_k = \mathrm{E}\left((X-\mu)^k\right)$ for $k \in \{2, 3, \ldots\}$.

(d) Compute the second raw moment of X, $\mu_2' = \mathrm{E}\left(X^2\right)$.

Solution to Question 1

Part (a): For f_X to be a p.d.f., it needs to (i) be nonnegative on $\mathbb{R}$ and (ii) integrate to 1. Part (i) is true since $x^2 \geq 0$ for all real x, and therefore $f_X(x) \geq 0$ for all real x. For (ii), recall that

$$\frac{\mathrm{d}}{\mathrm{d}x} \arctan x = \frac{1}{1+x^2}.$$

and

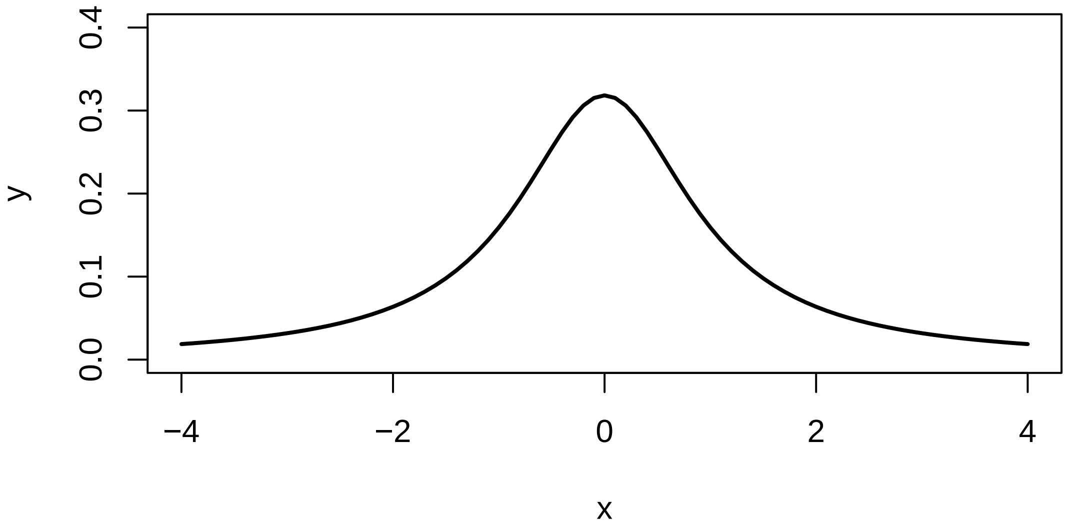
$$\lim_{\theta \to \frac{\pi}{2}} \tan\theta = \infty, \qquad \lim_{\theta \to -\frac{\pi}{2}} \tan\theta = -\infty.$$

Then,

$$\int_{-\infty}^{\infty} f_X(x)\,\mathrm{d}x = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{1}{(1+x^2)}\,\mathrm{d}x = \frac{1}{\pi}\left[\arctan x\right]_{-\infty}^{\infty} = \frac{1}{\pi}\left[\tfrac{\pi}{2} - \left(-\tfrac{\pi}{2}\right)\right] = 1.$$

One can take the first derivative to show that show that f_X has a local maximum at $(0, \frac{1}{\pi}) = (0, 0.318)$, and notice that $\lim_{x\to\infty} f_X(x) = \lim_{x\to-\infty} f_X(x) = 0$ to plot:

```
# enter the following commands in an R console:
x <- seq(from=-4, to=4, by=0.1)
y <- dcauchy(x)
plot(x, y, type='l', lwd=2, ylim=c(0, 0.4))
```



One could compute the second derivative to obtain the two inflection points $\left(\pm\frac{1}{\sqrt{3}}, \frac{3}{4\pi}\right) = (\pm 0.577, 0.239)$, but it is not essential to this exercise.

In order to see $\frac{\mathrm{d}}{\mathrm{d}x}\arctan x = \frac{1}{1+x^2}$, first recall the trigonometric identity:

$$\sin^2\theta + \cos^2\theta = 1 \qquad \Rightarrow \frac{\sin^2\theta}{\cos^2\theta} + \frac{\cos^2\theta}{\cos^2\theta} = \frac{1}{\cos^2\theta} \qquad \Rightarrow \tan^2\theta + 1 = \sec^2\theta,$$

And using implicit differentiation,

$$y = \arctan x \qquad \Rightarrow \tan y = x \qquad \Rightarrow \frac{\mathrm{d}}{\mathrm{d}x}\tan y = \frac{\mathrm{d}}{\mathrm{d}x}x \qquad \Rightarrow \sec^2 y\frac{\mathrm{d}y}{\mathrm{d}x} = 1 \qquad \Rightarrow \frac{\mathrm{d}y}{\mathrm{d}x} = \frac{1}{1+x^2}.$$

To find the derivative of $\tan\theta$, write $\tan\theta = \frac{\sin\theta}{\cos\theta}$ and use the quotient rule.

Part (b): Looking at the plot of the p.d.f., one would guess that $\mathrm{E}(X) = 0$. However, when one does the computation:

$$\mathrm{E}(X) = \int_{-\infty}^{\infty} x f_X(x)\,\mathrm{d}x = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x}{1+x^2}\,\mathrm{d}x = \frac{1}{\pi}\cdot\frac{1}{2}\left[\log\left(1+x^2\right)\right]_{-\infty}^{\infty}.$$

At this point it is tempting to say that since $\lim_{x\to\infty}\log(1+x^2) = \infty$ and $\lim_{x\to-\infty}\log(1+x^2) = \infty$ and so the two terms "cancel out" and the answer is 0, **but this is incorrect**.

In Section 1.1.6 it is mentioned that moments are defined 'long as the integral exists', and this is discussed in further detail in Appendix A.1. When computing moments and dealing with improper integrals, one must consider the two partial integrals in the right hand side of

$$\int_{-\infty}^{\infty} x f_X(x)\,\mathrm{d}x = \int_{-\infty}^{0} x f_X(x)\,\mathrm{d}x + \int_{0}^{\infty} x f_X(x)\,\mathrm{d}x,$$

(where for this example we split the integral at $a = 0$). Then in order for the integral on the left-hand side to exist, either:

- at least one of the integrals on the right-hand converges, or
- both integrals are ∞, or both are $-\infty$.

The reason for the second point is that we cannot have the situation where one term is ∞ and the other is $-\infty$; in that case the integral on the left-hand side ("$\infty - \infty$") is undefined.

However, this is precisely the situation we now have:

$$\int_{-\infty}^{0} x f_X(x)\,\mathrm{d}x = \frac{1}{\pi}\cdot\frac{1}{2}\left[\log\left(1+x^2\right)\right]_{-\infty}^{0} = -\infty,$$

$$\int_{0}^{\infty} x f_X(x)\,\mathrm{d}x = \frac{1}{\pi}\cdot\frac{1}{2}\left[\log\left(1+x^2\right)\right]_{0}^{\infty} = \infty.$$

Therefore, **for the Cauchy distribution, $\mu = \mathrm{E}(X)$ is undefined**.

Part (c): For $k \in \{2, 3, \dots\}$, the computation for the kth **central** moment involves adding/subtracting (a power of) the first moment μ, which is shown in Part (b) to be undefined. Therefore, the central moments are all undefined (for $k \geq 2$).

Part (d): Although the first moment is undefined, the second moment is defined, but infinite. One computes $\mathrm{E}\left(X^2\right)$ as

$$\frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x^2}{1+x^2}\,\mathrm{d}x = \frac{1}{\pi}\int_{-\infty}^{\infty}\left[1 - \frac{1}{1+x^2}\right]\mathrm{d}x = \frac{1}{\pi}\left[x - \arctan x\right]_{-\infty}^{\infty} = \frac{1}{\pi}\left[x\right]_{-\infty}^{\infty} - \frac{1}{\pi}\left[\arctan x\right]_{-\infty}^{\infty} = \infty - 1 = \infty.$$

Notice the difference between a moment being undefined and a moment being infinite.

Question 2

Suppose that the random variable X is known to only take non-zero values in the bounded range $[a, b]$, i.e. the support of X is $[a, b]$.

(a) Derive the expression $(X-a)(X-b) = \left(X - \frac{a+b}{2}\right)^2 - \frac{(b-a)^2}{4}$.

(b) Use the expression from (a) to prove that $\operatorname{Var}(X) \leq \dfrac{(b-a)^2}{4}$.

(c) Conclude that if $X \sim \operatorname{Bern}(p)$, for some $p \in [0,1]$, then $\operatorname{Var}(X) \leq \frac{1}{4}$.

(d) Is the bound $\operatorname{Var}(X) \leq \dfrac{(b-a)^2}{4}$ tight? In other words, is there a distribution F_X with support $[a, b]$ for which $\operatorname{Var}(X) = \dfrac{(b-a)^2}{4}$?

Solution to Question 2

Part (a): We first derive the expression $(X-a)(X-b) = \left(X - \frac{a+b}{2}\right)^2 - \frac{(b-a)^2}{4}$ by completing the square:

$$\begin{aligned}
(X-a)(X-b) &= X^2 - (a+b)X + ab \\
&= \left(X^2 - (a+b)X + \left(\tfrac{a+b}{2}\right)^2\right) + ab - \tfrac{(a+b)^2}{4} \\
&= \left(X - \tfrac{a+b}{2}\right)^2 + \tfrac{1}{4}\left[4ab - (a+b)^2\right] \\
&= \left(X - \tfrac{a+b}{2}\right)^2 + \tfrac{1}{4}\left[4ab - (a^2 + 2ab + b^2)\right] \\
&= \left(X - \tfrac{a+b}{2}\right)^2 + \tfrac{1}{4}\left[-(a^2 - 2ab + b^2)\right] \\
&= \left(X - \tfrac{a+b}{2}\right)^2 - \tfrac{1}{4}\left[(a-b)^2\right] \\
\Rightarrow (X-a)(X-b) &= \left(X - \tfrac{a+b}{2}\right)^2 - \tfrac{(b-a)^2}{4}.
\end{aligned}$$

Part (b) The expression proved in Part (a) implies

$$\left(X - \tfrac{a+b}{2}\right)^2 = (X-a)(X-b) + \tfrac{(b-a)^2}{4}. \qquad (1)$$

Recall Theorem 1.1.1: Given a random variable X, then over all values $c \in \mathbb{R}$,

$$\min_c \mathrm{E}[(X-c)^2] = \mathrm{E}[(X - \mathrm{E}[X])^2].$$

This implies that for any $c \in \mathbb{R}$,

$$\operatorname{Var}(X) = \mathrm{E}[(X - \mathrm{E}[X])^2] \leq \mathrm{E}[(X-c)^2].$$

In particular, using $c = \frac{a+b}{2}$, and Equation (1),

$$\operatorname{Var}(X) \leq \mathrm{E}[\left(X - \tfrac{a+b}{2}\right)^2] = \mathrm{E}[(X-a)(X-b) + \tfrac{(b-a)^2}{4}] = \mathrm{E}[(X-a)(X-b)] + \tfrac{(b-a)^2}{4}$$

since $\frac{(b-a)^2}{4}$ is a constant, and using the linearity of the expectation. Now, since $X \in [a, b]$, this implies

$$\begin{aligned}
&(X-a)(X-b) \leq 0, \\
\Rightarrow &\mathrm{E}[(X-a)(X-b)] \leq 0,
\end{aligned}$$

and therefore

$$\operatorname{Var}(X) \leq \mathrm{E}[(X-a)(X-b)] + \tfrac{(b-a)^2}{4} \leq \tfrac{(b-a)^2}{4}.$$

For **Part (c)**, since any Bernoulli random variable X only takes values of either 0 or 1, it itself is bounded in the range $[0,1]$, and the result follows. (Note: it has nothing to do with the value of p.)

For **Part (d)**, consider the discrete distribution F_X for the random variable X defined on $[a,b]$ by

$$X = \begin{cases} a, & \text{with probability } \frac{1}{2} \\ b, & \text{with probability } \frac{1}{2} \end{cases}.$$

Then

$$\begin{aligned} \mathrm{E}(X) &= \sum_x x\mathrm{P}(X=x) = a \cdot \mathrm{P}(X=a) + b \cdot \mathrm{P}(X=b) = \frac{a+b}{2} \\ \mathrm{E}\left(X^2\right) &= \sum_x x^2\mathrm{P}(X=x) = a^2 \cdot \mathrm{P}(X=a) + b^2 \cdot \mathrm{P}(X=b) = \frac{a^2+b^2}{2} \\ \mathrm{Var}\left(X^2\right) &= \mathrm{E}\left(X^2\right) - (\mathrm{E}(X))^2 = \frac{a^2+b^2}{2} - \left(\frac{a+b}{2}\right)^2 = \frac{(b-a)^2}{4} \end{aligned}$$

Therefore, the bound on the variance is tight, since this probability distribution has exactly this variance.

One might wonder what would inspire one to think of this example. First, one has to remember to work within the constraint that the support is $[a,b]$. Then, in order to maximise the variance, the idea (might) be to separate the probability mass function to place half of the mass at one extreme and the other half of the mass at the other extreme. The resulting thinking would lead one to the discrete distribution defined above. Of course, this is only an outline of one possible train of thought that would lead to this solution.

Question 3 (using R)

(a) Use R to generate 10 observations from a normal distribution with mean 3 and variance 2. Save the values in a vector `x`.

(b) Use the built-in R commands to compute the sample mean, variance and standard deviation of `x`.

(c) Write your own R functions to compute the sample mean and sample variance of `x`.

Solution to Question 3

Part (a):

The following commands can be entered in the R console:

```
set.seed(1)
x <- rnorm(n=10, mean=3, sd = sqrt(2))
print(x)
#>  [1] 2.114 3.260 1.818 5.256 3.466 1.840 3.689 4.044 3.814
#> [10] 2.568
```

Part (b):

Again, the following commands can be entered in the R console:

```
print(mean(x))
#> [1] 3.187

print(var(x))
#> [1] 1.219

print(sd(x))
#> [1] 1.104
```

Part (c):

The following functions can be created in the R console, but it is better to save them in an Rscript, e.g. a file named `myscript.R`:

```
# Note that it is also possible to use for-loops, but the
# built-in function `sum` is quite useful.

my_mean <- function(x){
    return(  sum(x) / length(x)  )
}

my_var <- function(x){
    return(   sum(  ( x - mean(x) )^2  )  /  (length(x) - 1)   )
}
```

Then the functions can be called from the R console:

```
# if saved in a script called "myscript.R", then uncomment and
# call the following command (one needs to `source` an Rscript before calling it):
#source("myscript.R")

print(my_mean(x))
#> [1] 3.187

print(my_var(x))
#> [1] 1.219
```

Question 4 (using R)

Suppose there is a file named `data1.txt` which contains the following data:

```
x,y
2,3
4,6
6,9
8,12
```

(Either download the file from Blackboard, or copy-paste the data into a file and name it `data1.txt`. **If you copy-paste, be sure to include an additional blank line after the line with '8, 22'.**)

(a) Use the function `read.table` to read the data from `data1.txt` into a data frame object named `df`.

(b) Extract a vector named `x`, containing values $(2, 4, 6, 8)$ from the data frame `df`. Similarly, extract a vector named `y`, containing values $(3, 6, 9, 12)$ from the data frame `df`.

(c) Create a vector named `z` which is the mean of the two vectors `x` and `y`, i.e. `z` contains four values, the first of which is $(2 + 3)/2 = 2.5$.

(d) Add the vector `z` to the data frame `df` so that `df` contains three columns, `x`, `y` and `z`.

(e) Write the data frame `df` to a file named `data2.txt`, so that this file contains:

Solution to Question 4

The following script contains all the lines needed to complete the exercise.

```r
q4 <- function(){
    # 5(a)
    df <- read.table("data1.txt", sep=",", header=T)

    # 5(b)
    x <- df$x
    y <- df$y

    # 5(c)
    z <- (x + y)/2

    # 5(d)
    df["z"] <- z

    # 5(e)
    write.table(df, file="data2.txt", col.names=T, row.names=F, quote=F, sep=",")
}
```

These lines can be run individually in the terminal, or the script (if it is saved as a file called `q4script.R`) can be called from the terminal using:

```
source("q4script.R")
q4()
```

Note there are alternatives for Questions 5(a) and 5(e):

```
# 5(a) alternative
df <- read.csv("ps8_data1.txt")

# 5(e) alternative
write.csv(df, "ps8_data2.txt", quote=F, row.names=F)
```

However, the reason that I suggest using the `read.table` command, rather than the `read.csv` command, is that learning to use `read.table` and how to set its parameters is more useful. For example, if instead of giving the data in `data1.txt` as above, one had to read the following data contained in a file named `data3.txt`:

```
2 3
4 6
6 9
8 12
```

Then, in this case, the file is not in csv format (the data are not comma-separated) and the `read.csv` command is not useful. Rather, one should use the commands:

```
# reads in the data
df3 <- read.table("data3.txt", sep=" ")

# sets the column names to be "x" and "y"
colnames(df3) <- c("x", "y")
```

R scripts and using the source function The solution presented above creates an R script named `q4script.R`, which contains the function `q4()`. Then, after using `source("q4script.R")`, one still needs to call the function `q4()`. The `source` function essentially makes R read all the lines in the script. But, why bother to create a function? Why not just have something such as the following in `q4script2.R`:

```
## q4script2.R
df <- read.table("data1.txt", sep=",", header=T)
df["z"] <- (df$x + df$y)/2
write.csv(df, file="data2.txt", quote=F, row.names=F)
```

and then once one calls `source("q4script2.R")`, it will run all the lines—surely this is just as good, and saves us a line calling the function `q4()`? In fact, this second option would work, but it is **bad practice**. The `source` function indeed reads in all the lines, and (a) if the line is inside a function, it checks if any of the lines doesn't make sense (e.g. there is a syntax error), or (b) any line not inside a function is executed by R.

If the last lines of `q4script.R` and `q4script2.R` both had the same syntax error

```
wrrrrritte.csv(df, file="data2.txt", quote=F, row.names=F)
```

then `source("q4script.R")` would identify the error without running the lines inside the function `q4()`, but `source("q4script2.R")` would execute the previous lines before identifying the error—if your script takes an hour to run, it isn't ideal to find out at the end of the hour that the last line contained an error!