STA120 Exercises 2

Jan Hohenheim

3/7/2022

Setup

```r
library("ggplot2")
library("ggthemes")
library("latex2exp")
library("cowplot")
```

```
##
## Attaching package: 'cowplot'
## The following object is masked from 'package:ggthemes':
##
##     theme_map
```

```r
library("qqplotr")
```

```
##
## Attaching package: 'qqplotr'
## The following objects are masked from 'package:ggplot2':
##
##     stat_qq_line, StatQqLine
```

```r
library("glue")
```

Problem 4 (Discrete uniform distribution)

Let m be a positive integer. The discrete uniform distribution is defined by the pmf P(X = k) = 1/m , k = 1, . . . , m. ### (a) Visualize the pmf and cdf of a discrete uniform random variable with m = 6.

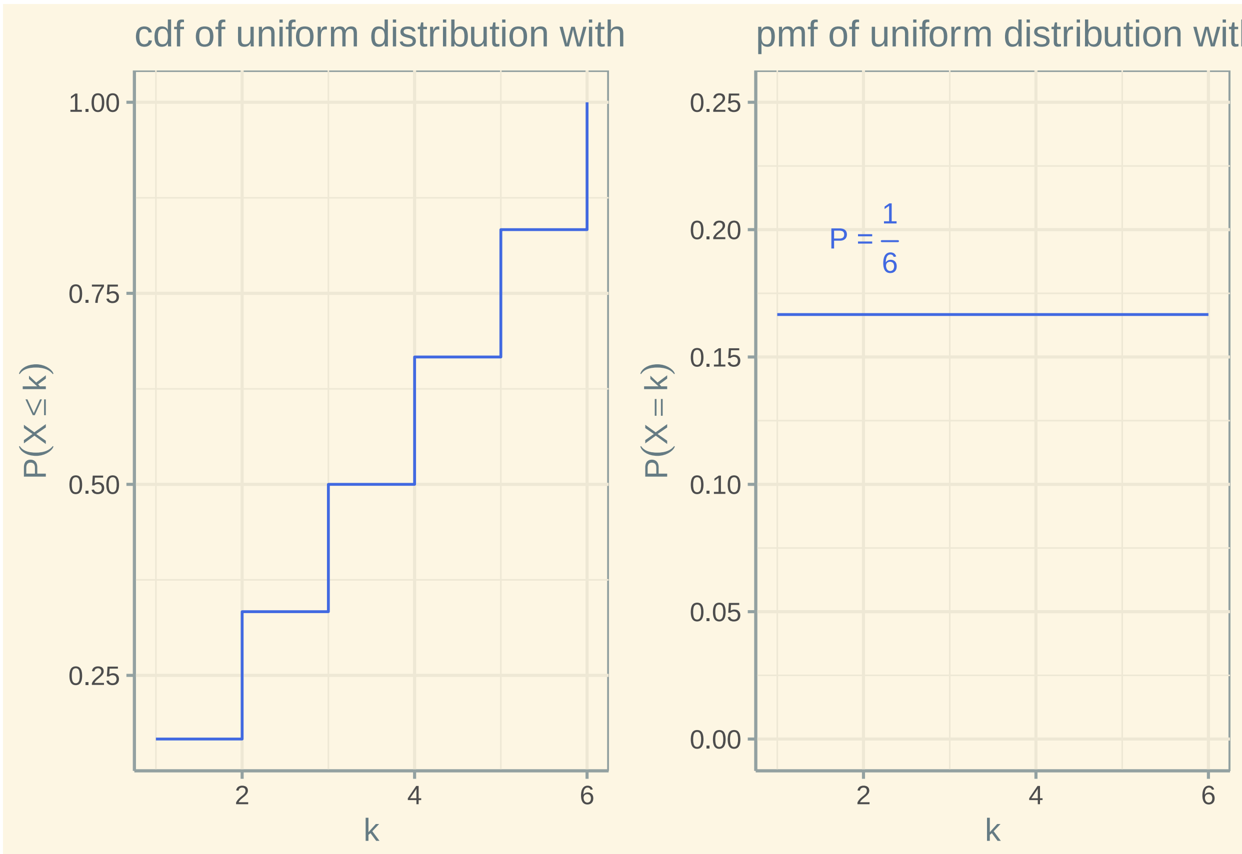
```r
m <- 6
step_data <- data.frame(k = 1:m)
plot_cdf <- step_data |>
  ggplot(aes(x = k)) +
  stat_ecdf(geom = "step", pad = FALSE, color = "royalblue") +
  labs(title = "cdf of uniform distribution with m = 6", x = "k" , y = TeX("$P(X \\leq k)$")) +
  theme_solarized()

probabilities = data.frame(k = 1:m, p = rep(1/m, m))
plot_pmf <- probabilities |>
  ggplot(aes(x = k, y = p)) +
  geom_line(color = "royalblue") +
  annotate("text", x = 2, y = (1/6 + 0.03), label = TeX("P = \\frac{1}{6}"), color = "royalblue") +
  labs(title = "pmf of uniform distribution with m = 6", x = "k", y = TeX("$P(X = k)$")) +
  ylim(0, 0.25) +
```

```
  theme_solarized()

plot_grid(plot_cdf, plot_pmf)
```

```
## Warning in is.na(x): is.na() applied to non-(list or vector) of type
## 'expression'
```

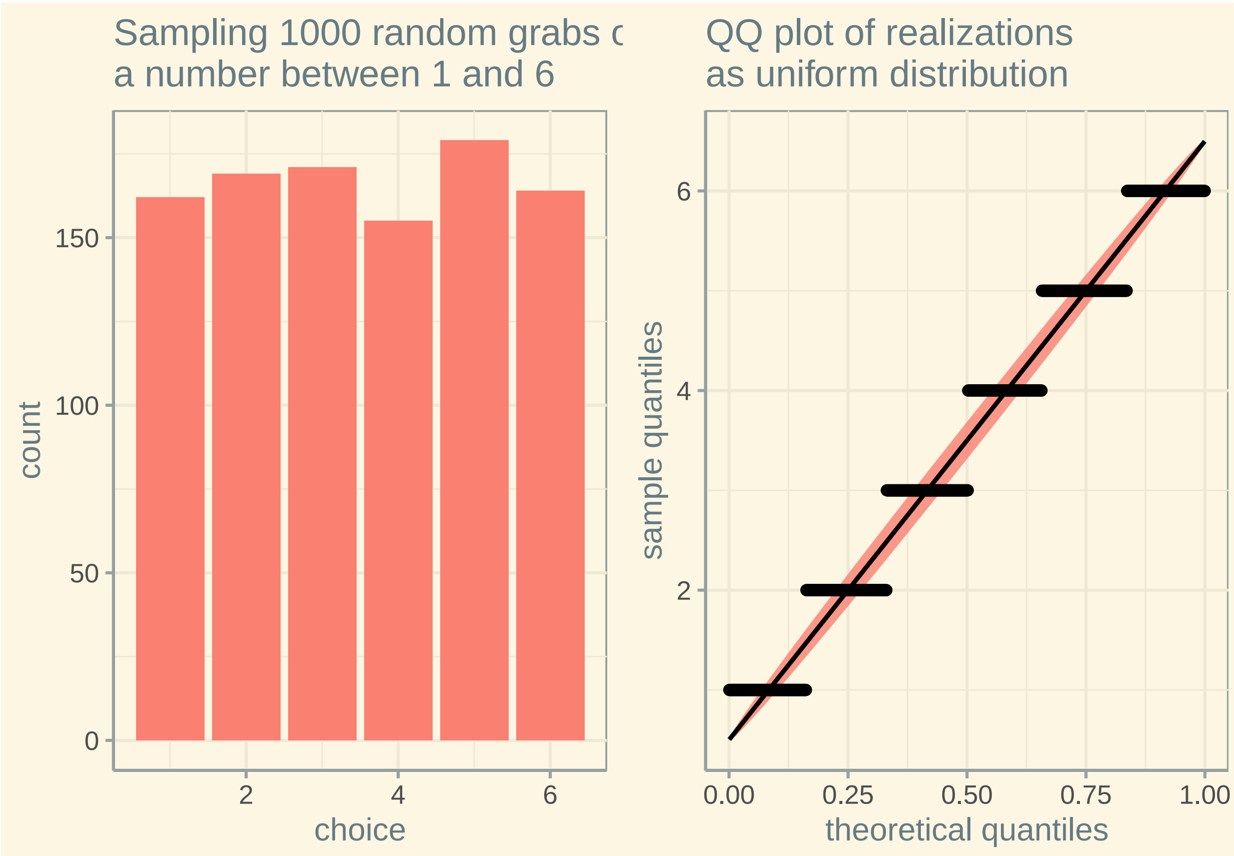


(b) Draw several realizations from X, visualize the results and compare to the pmf of a). What are sensible graphics types for the visualization? Hint: the function sample() conveniently draws random samples without replacement.

```
realizations <- data.frame(x = sample(1:6, 1000, replace = TRUE))
plot_realizations <- realizations |>
  ggplot(aes(x)) +
  geom_bar(fill = "salmon") +
  labs(title = "Sampling 1000 random grabs of\na number between 1 and 6", x = "choice", y = "count") +
  theme_solarized()

di <- "unif"
qq_plot <- ggplot(realizations, aes(sample = x)) +
  stat_qq_band(distribution = di, fill = "salmon") +
  stat_qq_line(distribution = di) +
  stat_qq_point(distribution = di) +
  labs(title = "QQ plot of realizations\nas uniform distribution", x = "theoretical quantiles", y = "sa
  theme_solarized()

plot_grid(plot_realizations, qq_plot)
```



(c) Compute E(X) and Var(X)

```
values <- 1:6
probabilities <- 1/6
e <- sum(values * probabilities)
glue("E(X) = {e}")
```

```
## E(X) = 3.5
```

```
squared_deviations_from_mean = (values - e)^2
variance <- sum(squared_deviations_from_mean * probabilities)
glue("Var(X) = {variance}")
```

```
## Var(X) = 2.91666666666667
```

Problem 5 (Uniform distribution)

We assume X ~ U(0, θ), for some value $R \ni \theta > 0$. Hint: The density of X is given by f (x) = 1/ θ for $0 \leq x \leq \theta$ and f (x) = 0 otherwise.

(a) For all a, b $\in$ R with 0 < a < b < θ show that P(X $\in$ [a, b]) = (b-a)/θ

$P(X \in [a,b]) = \sum_{i=a}^{b} P(X = i) = \sum_{i=a}^{b} i\frac{1}{\theta} = \frac{b-a}{\theta}$ QED

(b) Calculate E(X), SD(X) and the quartiles of X

E(X) $E(X) = \sum_{i=0}^{\theta} iP(i) = \sum_{i=0}^{\theta} i\frac{1}{\theta} = \frac{1}{\theta}\sum_{i=0}^{\theta} i$ Per Gauss: $\frac{1}{\theta}\sum_{i=0}^{\theta} i = \frac{1}{\theta}\frac{\theta(\theta+1)}{2} = \frac{\theta+1}{2}$

SD(X) $SD(X) = \sqrt{Var(X)}$
$Var(X) = E((X-\mu)^2) = \frac{1}{\theta}\sum_{i=0}^{\theta}(i-\mu)^2 = \frac{1}{\theta}\sum_{i=0}^{\theta}(i-\frac{\theta+1}{2})^2 = \frac{1}{\theta}\sum_{i=0}^{\theta}(\frac{2i-\theta+1}{2})^2 = \frac{1}{\theta}\sum_{i=0}^{\theta}\frac{(2i-\theta+1)^2}{4}$
Sorry, but I don't know how to further simplify this :(

Quartiles Since the distribution is symmetric, the median is the mean. Further, the 1. quartile must be half of the median and the 3. quartile 1.5 times the median for the same reason. Thus:
- **1. quartile**: $\frac{\theta+1}{4}$
- **2. quartile**: $\frac{\theta+1}{2}$
- **3. quartile**: $3\frac{\theta+1}{4}$

Problem 6 (Calculating probabilities)

In the following settings, compute the probabilities and quantiles q1 and q2 using the R commands pnorm(a) and qnorm(b) for specific values a and b.
X ~ N(2, 16)

```
p1 <- pnorm(q = 4, mean = 2, sd = 4)
glue("P(X < 4) = {p1}")
```

```
## P(X < 4) = 0.691462461274013
```

```
p2 <- pnorm(q = 4, mean = 2, sd = 4) - pnorm(q = 0, mean = 2, sd = 4)
glue("P(0 <= X <= 4) = {p2}")
```

```
## P(0 <= X <= 4) = 0.382924922548026
```

```
q1 <- qnorm(p = 0.95, mean = 2, sd = 4, lower.tail = FALSE)
glue("P(X > q1) = 0.95 implies q1 = {q1}")
```

```
## P(X > q1) = 0.95 implies q1 = -4.57941450780589
```

```
q2 <- -qnorm(p = 0.05, mean = 2, sd = 4)
glue("P(X < -q2) = 0.05 implies q2 = {q2} = -q1")
```

```
## P(X < -q2) = 0.05 implies q2 = 4.57941450780589 = -q1
```