Jacob Snyder
Section 8

ACM116 Set 2

1. $Cov(X,Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$

$\mathbb{E}[XY] = P(\alpha < Z < \beta)(1) = \beta - \alpha$

$\mathbb{E}[X] = (\beta)(1) = \beta$

$\mathbb{E}[Y] = (1-\alpha)(1) = 1-\alpha$

$\rightarrow Cov(X,Y) = \beta - \alpha - \beta(1-\alpha) = \beta - \beta - \alpha + \alpha\beta = \boxed{\alpha(\beta-1)}$

2. $X+Y \sim \mathcal{N}(2\mu, 2\sigma^2)$ $\quad X-Y \sim \mathcal{N}(0, 2\sigma^2)$

$M_{X+Y}(t_1) = \exp(2\mu t_1 + \frac{2\sigma^2 t_1^2}{2})$ $\quad M_{X-Y}(t_2) = \exp(\frac{2\sigma^2 t_2^2}{2})$

$M_{(X+Y)(X-Y)}(t_1,t_2) = \exp(\sum_{i=1}^{2} t_i \mu_i + \frac{1}{2}\sum_{i=1}^{2} t_i^2 \mathbb{V}[X_i] + \sum_{i \neq j} t_i t_j Cov(X_i, X_j))$

Note that $Cov(X+Y, X-Y) = \mathbb{E}[(XX-YY)] - \mathbb{E}[X+Y]\mathbb{E}[X-Y] = 0 - 0 = 0$

$= \exp(2\mu t_1 + \frac{1}{2}(2t_1^2\sigma^2 + 2t_2^2\sigma^2))$

$= M_{(X+Y)}(t_1) M_{(X-Y)}(t_2) \rightarrow (X+Y)$ and $(X-Y)$ are independent.

3. a) $Y \sim \mathcal{N}(0,1)$. This is because $\mathcal{N}(0,1)$ is centered about 0, with identical probability distributions on either side of 0. This means that sometimes multiplying by -1 effectively has no effect on the probability distribution. Therefore the distribution of Y is the same as that of X, that is, $\mathcal{N}(0,1)$.

b) $Cov(X,Y) = \mathbb{E}[XY] - \mathbb{E}(X)\mathbb{E}(Y) = \mathbb{E}[XY] = \mathbb{E}[XXZ] = 0$

So X and Y are not correlated.

c) Suppose that $X \perp\!\!\!\perp Y$. Then:

$\mathbb{P}(X \in [-1,1], Y \in [-1,1]) = \mathbb{P}(X \in [-1,1])\mathbb{P}(Y \in [-1,1])$

$\rightarrow \mathbb{P}(X \in [-1,1]) = \mathbb{P}(X \in [-1,1])\mathbb{P}(Y \in [-1,1])$ because if $X \in [-1,1]$ then $Y \in [-1,1]$

$\rightarrow 1 = \mathbb{P}(Y \in [-1,1])$. So we have reached a contradiction, so X and Y are not independent.

4. a) By the Markov Inequality, $\mathbb{P}(S_n \geq m) \leq \frac{\mathbb{E}[S_n]}{m}$

$\mathbb{E}[S_n] = \mathbb{E}[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} \mathbb{E}[X_i] = n\lambda$

$\rightarrow \boxed{\mathbb{P}(S_n \geq m) \leq \frac{n\lambda}{m}}$

b) For large enough n, $\bar{X}_n \sim \mathcal{N}(\lambda, \frac{\lambda^2}{n})$, $\frac{\bar{X}_n - \lambda}{\lambda/\sqrt{n}} \sim \mathcal{N}(0,1)$

$\mathbb{P}(S_n \geq m) = \mathbb{P}(\bar{X}_n \geq \frac{m}{n}) = \mathbb{P}\left(\frac{\bar{X}_n - \lambda}{\lambda/\sqrt{n}} \geq \frac{m/n - \lambda}{\lambda/\sqrt{n}}\right) \boxed{1 - \Phi\left(\frac{m/n - \lambda}{\lambda/\sqrt{n}}\right)}$

c) MATLAB The Markov bound is not very tight.

5. MATLAB b) We get 0.25 every time because $g(x)/f(x) = \frac{x^3}{f(x|4,1)} = \frac{x^3}{\frac{\Gamma(5)}{\Gamma(4)} x^3} = \boxed{\frac{1}{4}}$ so it's a constant, not an R.V.

6. MATLAB

```
grthanct = 0;
for a = 1:10000
    data = poissrnd(1,100,1);
    datasum = sum(data);
    if datasum >= 120
        grthanct = grthanct + 1;
    end
end

markovbound = 100./120;

cltval = (12./10 - 1)./(1./10);
cltprob = 1 - normcdf(2,0,1);
rlprob = grthanct./10000;

disp("Markov Bound: ");
disp(markovbound);

disp("CLT estimate: ");
disp(cltprob);

disp("Estimated probability: ");
disp(rlprob);
```

```
Markov Bound:
    0.8333

CLT estimate:
    0.0228

Estimated probability:
    0.0241
```

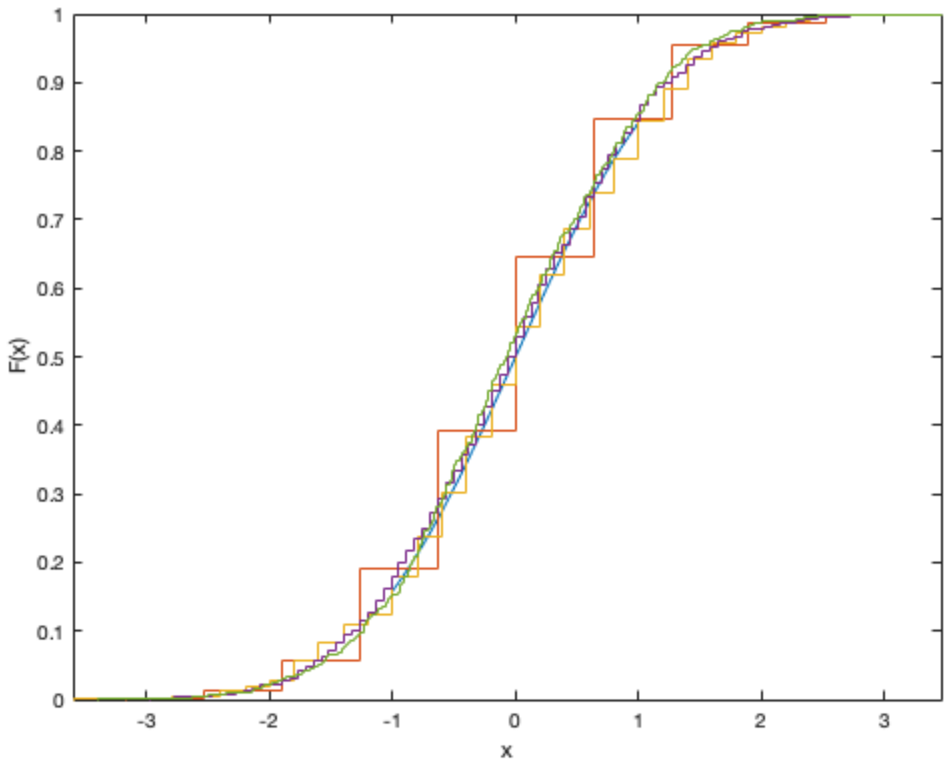
Published with MATLAB® R2018a

```
data = rand(100, 1);
data2 = data.^3;
mn = mean(data2);

betadata = betarnd(4,1);
betamn = betadata.^3;
betadiv = betapdf(betadata,4,1);

disp("Monte Carlo estimate: ");
disp(mn);

disp("Beta Monte Carlo estimate: ");
disp(betamn./betadiv);
```

```
Monte Carlo estimate:
    0.2336

Beta Monte Carlo estimate:
    0.2500
```

Published with MATLAB® R2018a

```
Ydata10 = [];
Ydata100 = [];
Ydata1000 = [];
Ydata10000 = [];

for a = 1:1000
    data = rand(10,1);
    data(data>0.5) = 1;
    data(data<=0.5) = -1;
    Ydata10 = [Ydata10 sqrt(10).*mean(data)];

    data = rand(100,1);
    data(data>0.5) = 1;
    data(data<=0.5) = -1;
    Ydata100 = [Ydata100 sqrt(100).*mean(data)];

    data = rand(1000,1);
    data(data>0.5) = 1;
    data(data<=0.5) = -1;
    Ydata1000 = [Ydata1000 sqrt(1000).*mean(data)];

    data = rand(10000,1);
    data(data>0.5) = 1;
    data(data<=0.5) = -1;
    Ydata10000 = [Ydata10000 sqrt(10000).*mean(data)];
end

fplot(@(x) normcdf(x), [-1,1]);
hold on;
ecdf(Ydata10);
ecdf(Ydata100);
ecdf(Ydata1000);
ecdf(Ydata10000);
```



Published with MATLAB® R2018a