HYD 142 Exam

Kenneth Larrieu

1.

a. If $Z = X - Y$, where X and Y are independent continuous random variables, then the distribution of Z can be computed as the convolution of X and $-Y$. More precisely:

$$\boxed{f_Z(z) = \int_{-\infty}^{\infty} f_X(z+y) f_Y(y) dy}$$

b. If X and Y are dependent continuous random variables, then a conditional distribution must be used in the integral:

$$\boxed{f_Z(z) = \int_{-\infty}^{\infty} f_{X|Y}(z+y|y) f_Y(y) dy = \int_{-\infty}^{\infty} f_{Y|X}(x-z|x) f_X(x) dx}$$

c. If X and Y are independent discrete random variables:

$$\boxed{p_Z(z) = \sum_{y_i \in supp(Y)} p_X(z+y_i) p_Y(y_i)}$$

Similarly, if X and Y are dependent discrete random variables:

$$\boxed{p_Z(z) = \sum_{y_i \in supp(Y)} p_{X|Y}(z+y|y) p_Y(y) = \sum_{x_i \in supp(X)} p_{Y|X}(x-z|x) p_X(x)}$$

d. To get the characteristic function of Z for independent continuous X and Y we use the following property of a characteristic function: if $X_1, X_2, ...X_n$ are independent, then

$$\varphi_{a_1X_1+a_2X_2+...+a_nX_n}(t) = \varphi_{X_1}(a_1t) + \varphi_{X_2}(a_2t) + ... + \varphi_{X_n}(a_nt)$$

Thus,

$$\varphi_Z(t) = \varphi_X(t) + \varphi_Y(-t) = \mathbb{E}[e^{itX}] + \mathbb{E}[e^{-itY}] = \boxed{\int_{-\infty}^{\infty} f_X(x)e^{itx}dx + \int_{-\infty}^{\infty} f_Y(y)e^{-ity}dy}$$

e. To get the variance of Z, we can use the fact that $Var(Z) = Cov(Z, Z) = Cov(X-Y, X-Y)$ and that the covariance operator is bilinear:

$$Var(Z) = Cov(X-Y, X-Y) = Cov(X, X) + Cov(Y, Y) - 2Cov(X, Y) = \boxed{\sigma_X^2 + \sigma_Y^2 - 2Cov(X, Y)}$$

f. If $W = \max(X, Y)$ and X and Y are independent, then:

$$\boxed{F_W(w) = F_X(w) \cdot F_Y(w)}$$

g. If Y is a normal random variable, then $X = e^Y$ has a lognormal distribution. Then the pdf of X is defined as:

$$\boxed{f_X(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}}, \quad x \geq 0}$$

where μ and σ are the mean and standard deviation of the normal variable Y.

The gamma distribution has a pdf of the form:

$$f_X(x) = \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)}, \quad x \geq 0$$

where α and β are positive parameters. The gamma distribution can be thought of as a generalization of an exponential random variable, and in fact the sum of independent exponential random variables has a gamma distribution.

Both the lognormal and gamma distributions are similar in that they assume positive values, they are positively skewed, and they have similar applications. Both are used to model similar processes such as waiting time for an event of a certain magnitude.

h. The key difference between extreme distributions of type I and type II is that a type I distribution is defined for all real numbers, whereas a type II distribution has a lower bound for a maximum value distribution or an upper bound for a minimum value distribution.

i. A power law distribution is a proability distribution whose tail falls off as a power law (slower than an expoential). More precisely, the tail of the distribution (i.e. for large x) has the behavior:

$$f_X(x) = cx^{-\alpha}, \quad \alpha > 1$$

Such a distribution can be used to model the frequency of extreme flood events or earthquakes above a given magnitude. Experimentally, power-law distributions can provide a better fit to some data sets than Gumbel, Log-Pearson or other extreme value distributions.

j. Geometric distributions and Bernoulli distributions describe the same process, whereby experiments are conducted which may have one of two outcomes. A Bernoulli r.v. denotes the outcome of a single trial, whereas a geometric distribution denotes the number of trials until the first success. Typically p is used as a parameter to indicate the probability of success for a single trial.

Exponential and Poisson distributions both describe the same process, namely a Poisson process, whereby an event may occur or not occur at any point in time (i.e. along a continuum), with the probability of a event occuring within an interval being proportional to the size of the interval. An exponential distribution describes the amount of time before the first event occurs, whereas a Poisson distribution describes the number of events that occur within some amount of time. Typically λ is the parameter used to indicate the average rate of event occurences.

2.

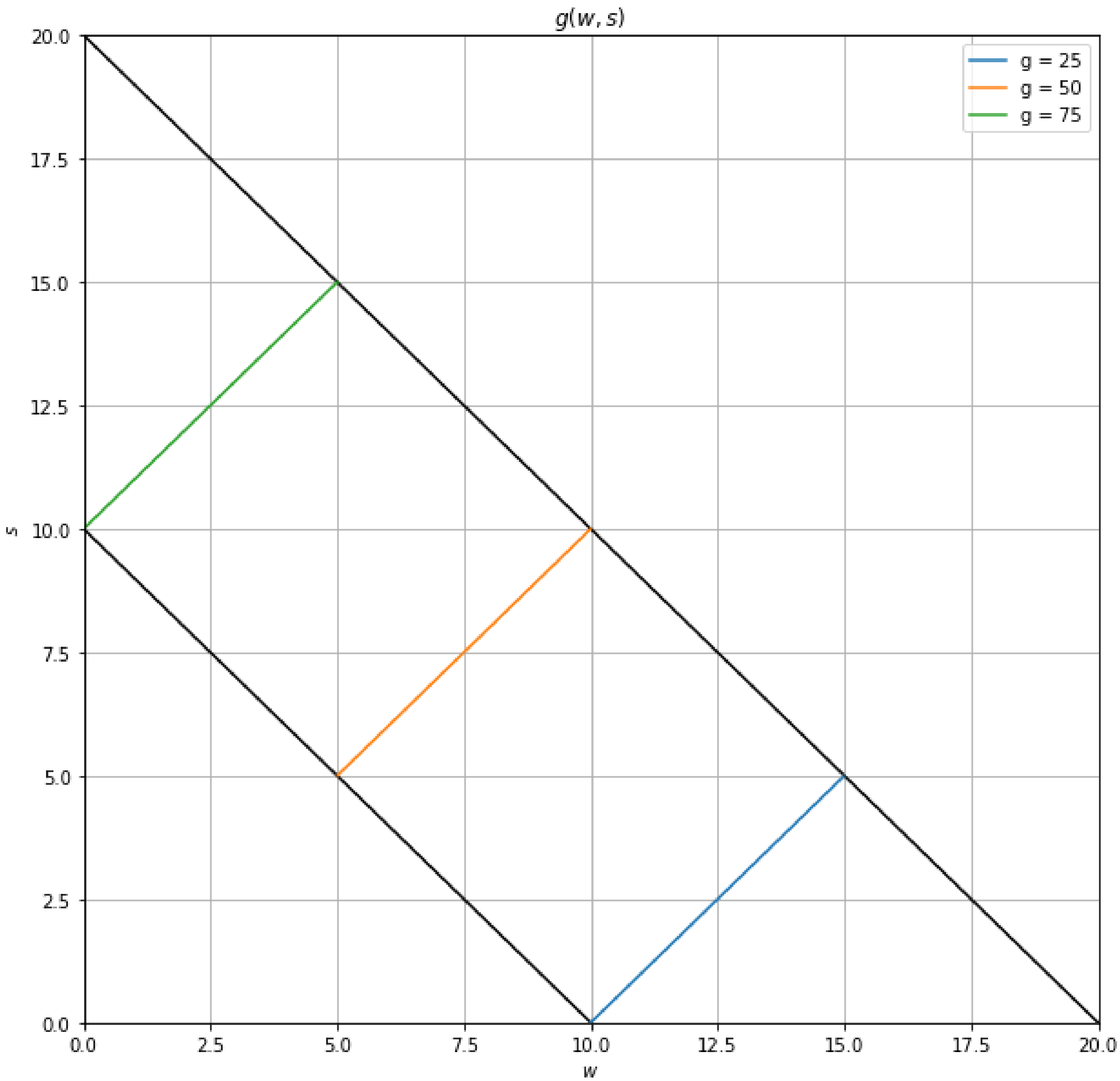
a. Plotting contours of constant g:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import scipy.special
%matplotlib inline

fig, ax = plt.subplots(1, 1, figsize=(10, 10))
xs = np.linspace(0, 21, 2001)
```

```
lb = [10 - x for x in xs]
ub = [20 - x for x in xs]
ax.plot(xs, lb, 'black')
ax.plot(xs, ub, 'black')
gs = range(25, 76, 25)
for g in gs:
    xc = [x for x in xs if 10 <= (g-50)/2.5 + 2*x <= 20]
    contour = [(g-50)/2.5 + x for x in xc]
    ax.plot(xc, contour, label=r'g = %s' % g)

ax.set_title(r'$g(w, s)$')
ax.set_xlabel(r'$w$')
ax.set_ylabel(r'$s$')
ax.set_xlim(0, 20)
ax.set_ylim(0, 20)
ax.grid()
ax.legend()
plt.show()
```



b. Since the $f_{W,S}(w,s) = K = \text{const.}$ over the trapezoid, the integral is simply:

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{W,S}(w,s)\, dw\, ds = K \cdot 150 = 1 \quad \rightarrow \quad \boxed{K = \frac{1}{150}}$$

c. From the plot above, we see that $G \geq 75$ corresponds to the integral of $f_{W,S}(w,s)$ over the area above the $g = 75$ contour and bounded by the black lines. We can compute this integral geometrically as:

$$P(G \geq 75) = K \cdot 25 = \frac{25}{150} = \boxed{\frac{1}{6}}$$

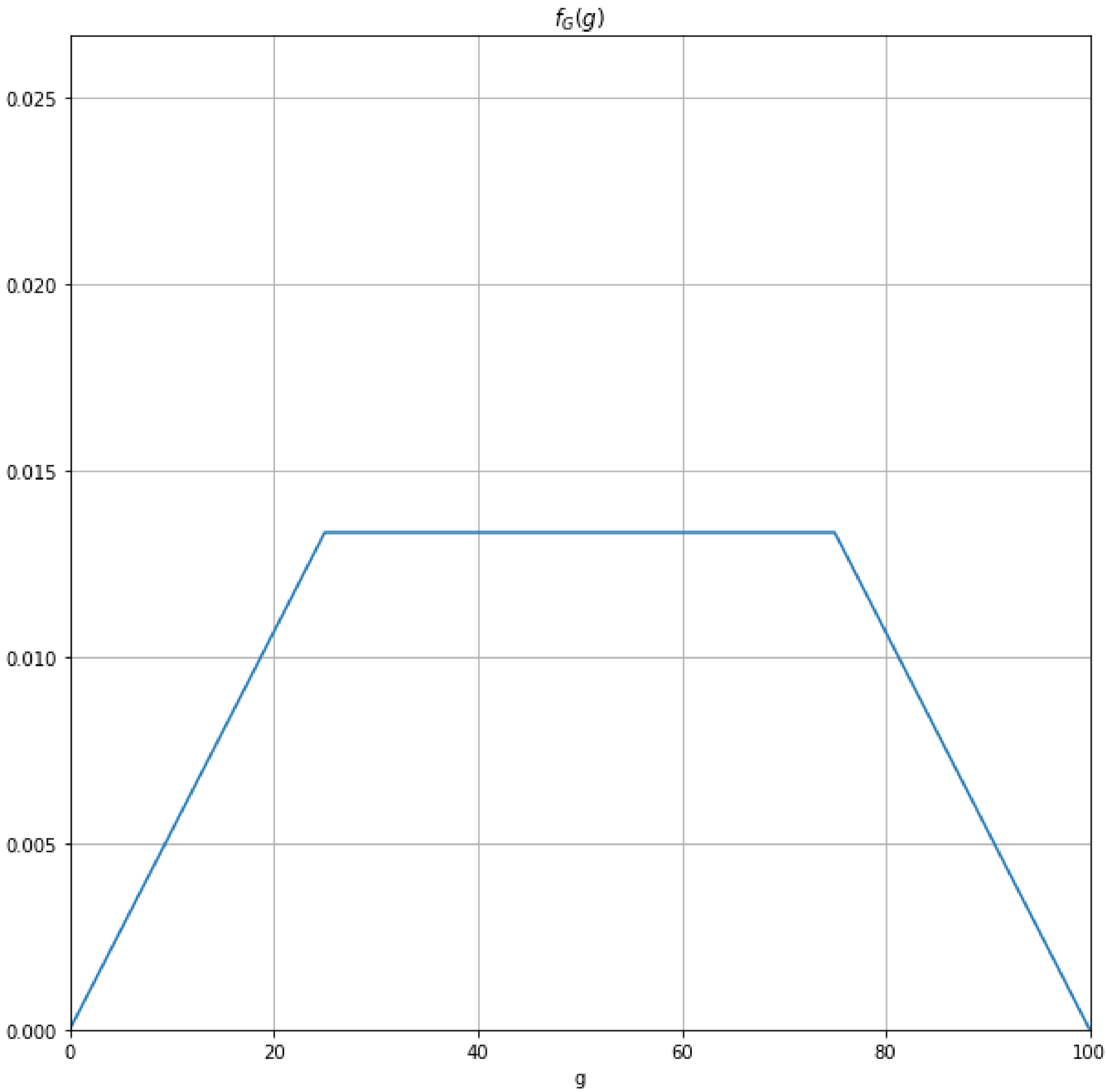
d. Integrating the joint density $F_{W,S}(w,s)$ over the area below the contour for g and within the bounds gives the CDF:

Integrating yields:

$$f_G(g) = \begin{cases} \frac{1}{75} \cdot \frac{g}{25} & 0 \leq g \leq 25 \\ \frac{1}{75} & 25 \leq g \leq 75 \\ \frac{40 - \frac{2}{5}g}{750} & 75 \leq g \leq 100 \\ 0 & \text{o.w.} \end{cases}$$

Graphing the pdf:

```
fig, ax = plt.subplots(1, 1, figsize=(10, 10))
g = np.linspace(0, 100, 101)
pdf = np.piecewise(g, [(g>=0)&(g<=25), (g>25)&(g <= 75), (g>75)&(g<=100)], [lambda g:
↪  g*1.0/(75*25), lambda g: 1.0/75, lambda g: (40 - g *2.0/5)/750.0])
ax.set_xlabel('g')
ax.set_title(r'$f_G(g)$')
ax.grid()
ax.set_xlim(0, 100)
ax.set_ylim(0, 2.0/75)
ax.plot(g, pdf)
plt.show()
```



e. Determine $P(W < 1|G = 75)$

$$P(W < 1|G = 75) = \frac{P(W < 1 \cap G = 75)}{P(G = 75)}$$

Since this probability corresponds to the portion of the $g = 75$ contour where $w < 1$, and the probability density is constant along this line, it is just the ratio of the line length left of $w = 1$ to the total line length. Thus, $\boxed{P(W < 1|G = 75) = \frac{1}{5}}$

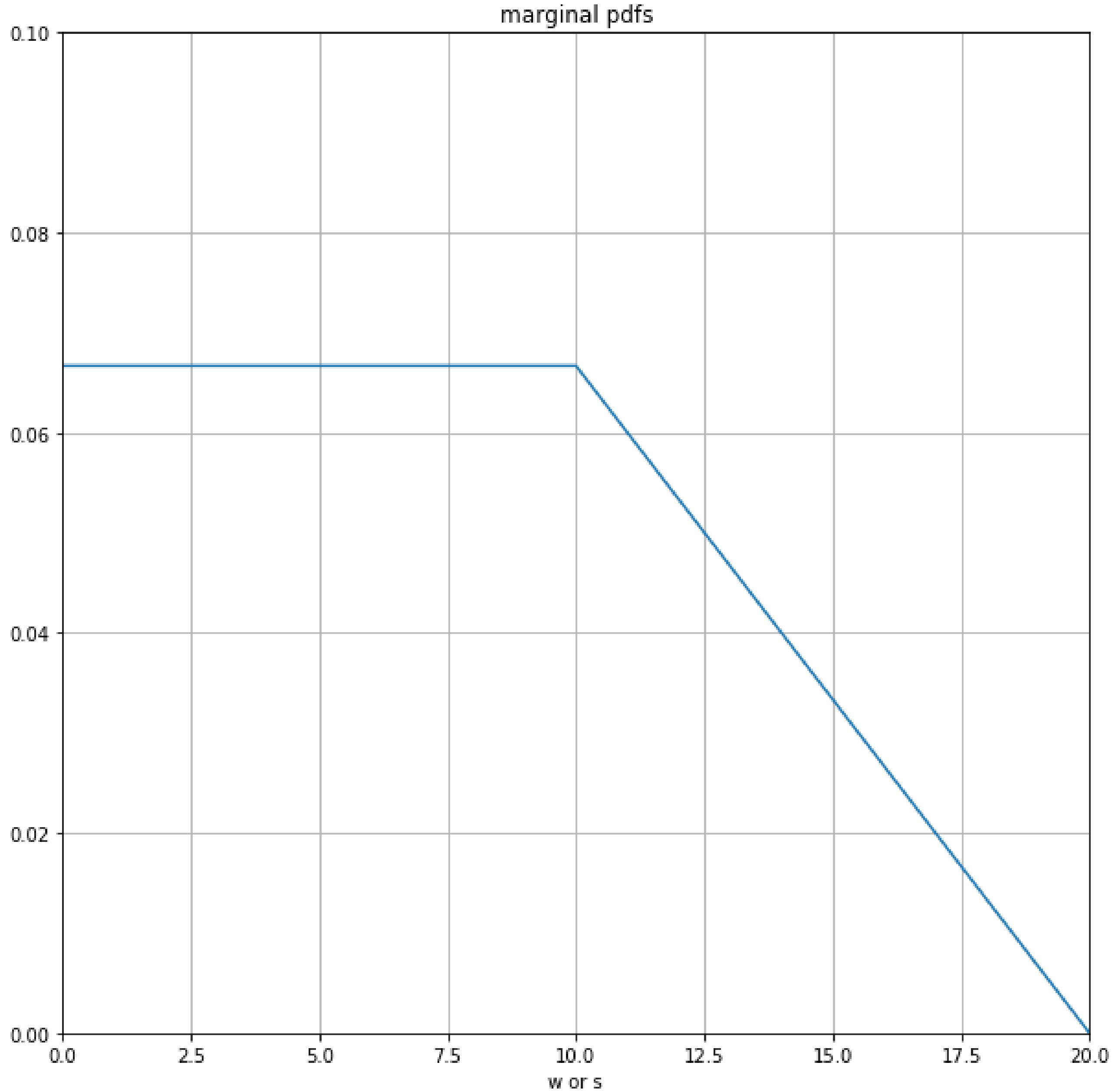
f.

$$f_W(w) = \int_{-\infty}^{\infty} f_{W,S}(w,s)ds = \begin{cases} \frac{1}{15} & 0 \leq w \leq 10 \\ \frac{20-w}{150} & 10 \leq w \leq 20 \\ 0 & \text{o.w.} \end{cases}$$

$$f_S(s) = \int_{-\infty}^{\infty} f_{W,S}(w,s)dw = \begin{cases} \frac{1}{15} & 0 \leq s \leq 10 \\ \frac{20-s}{150} & 10 \leq s \leq 20 \\ 0 & \text{o.w.} \end{cases}$$

These marginal densities are the same function. An illustration:

```
fig, ax = plt.subplots(1, 1, figsize=(10, 10))
x = np.linspace(0, 20, 101)
pdf = np.piecewise(x, [x <= 10, x > 10], [lambda x: 1.0/15, lambda x: (20-x)/150.0])
ax.set_xlabel('w or s')
ax.set_title('marginal pdfs')
ax.grid()
ax.set_xlim(0, 20)
ax.set_ylim(0, 0.1)
ax.plot(x,pdf)
plt.show()
```



g. Since the marginal densities are the same function, they will have the same mean and variance, so we only need to compute them once:

$$\mu_W = \mu_S = \int_0^{10} x \cdot \frac{1}{15} dx + \int_{10}^{20} x \cdot \frac{20-x}{150} dx = \frac{x^2}{30}\Big|_0^{10} + \frac{1}{150}\left[10x^2 - \frac{x^3}{3}\right]\Big|_{10}^{20} = \boxed{\frac{70}{9} \approx 7.778}$$

$$\sigma_W^2 = \sigma_S^2 = \mathbb{E}[X^2] - \mathbb{E}[X]^2 = \int_0^{10} x^2 \frac{1}{15} dx + \int_{10}^{20} x^2 \frac{20-x}{150} dx - \mu^2$$

$$\sigma^2 = \frac{x^3}{45}\Big|_0^{10} + \frac{1}{150}\left[\frac{20}{3}(7,000) - \frac{1}{4}(150,000)\right]\Big|_{10}^{20} - \left(\frac{70}{9}\right)^2$$

$$\boxed{\sigma^2 = \frac{1850}{81} \approx 22.84}$$

h. Clearly it is the case that $f_W(w) \cdot f_S(s) \neq f_{W,S}(w,s)$ (e.g. where $w > 10$, where $s > 10$, where $s < 10 - w$, or where $s > 20 - w$). Therefore, sleeping and working are not independent random variables.

3.

a. The nuts per cookie are is a Poisson random variable (call it X) with $\lambda = 1.5$. Thus,

$$P(X \geq 1) = 1 - P(X < 1) = 1 - P(X = 0) = \boxed{1 - \frac{1.5^0 e^{-1.5}}{0!} = 1 - e^{-1.5} \approx 0.78}$$

b.

$$\sigma_X^2 = \lambda = \boxed{1.5}$$

c. For a box of m cookies, then the number of nuts is equal to the sum of the nuts in each cookie X_i: call $C_m = X_1 + X_2 + ...X_m$, which is the total number of nuts in a box of m cookies. Similarly, say $C_n = X_1 + X_2 + ...X_n$. Due to the linearity of the expectation value operator:

$$\mathbb{E}[C_n] = \mathbb{E}[X_1] + \mathbb{E}[X_2] + ... + \mathbb{E}[X_n] = n \cdot \lambda = 1.5n$$

Since the number of cookies needs to be a whole number, we can see that the $C_m \neq \mathbb{E}[C_n]$ whenever n is odd, so $P(C_m = \mathbb{E}[C_n]) = 0$ in this case.

Then, in the case where n is even, we want $P(C_m = \mathbb{E}[C_n]) = P(C_m = 1.5n)$. The rest of the cases only apply to even n.

For $m = 1$, then $C_m = X$ and we have

$$p_{C_m}(c) = \frac{1.5^c e^{-1.5}}{c!}$$

In which case $P(C_m = \mathbb{E}[C_n]) = p_{C_m}(1.5n) = \frac{1.5^{1.5n} e^{-1.5}}{(1.5n)!}$

For $m = 2$, then $C_m = X_1 + X_2$, and

$$p_{C_m}(c) = P(X_1 + X_2 = c) = P(X_2 = c - X_1) = \sum_{x=0}^{c} p_{X_1}(x) p_{X_2}(c - x) = \sum_{x=0}^{c} \frac{1.5^c e^{-3}}{x!(c-x)!} = \frac{1.5^c e^{-3} 2^c}{c!}$$

In which case $P(C_m = \mathbb{E}[C_n]) = p_{C_m}(1.5n) = \frac{1.5^{1.5n} e^{-3} 2^{1.5n}}{(1.5n)!}$

For $m = 3$, then $C_m = X_1 + X_2 + X_3$, and

$$p_{C_m}(c) = P(X_3 = c - (X_1 + X_2)) = \sum_{x=0}^{c} p_{X_1+X_2}(x) p_{X_3}(c - x) = \sum_{x=0}^{c} \frac{1.5^c e^{-4.5} 2^x}{x!(c-x)!} = 1.5^c e^{-4.5} \frac{3^c}{c!}$$

In which case $P(C_m = \mathbb{E}[C_n]) = p_{C_m}(1.5n) = 1.5^{1.5n} e^{-4.5} \frac{3^{1.5n}}{(1.5n)!}$

From here we can see a pattern and generalize:

$$p_{C_m}(c) = 1.5^c e^{-1.5m}\frac{m^c}{c!} = \frac{(1.5m)^c e^{-1.5m}}{c!}$$

Thus, we can conclude that:

$$\boxed{P(C_m = \mathbb{E}[C_n]) = \begin{cases} \frac{(1.5m)^{1.5n}e^{-1.5m}}{(1.5n)!} & n = \text{even} \\ 0 & n = \text{odd} \end{cases}}$$

d. The probability of a randomly chosen nut is in a cookie containing K nuts is proportional to the probability that a cookie contains K nuts times K, since there are K nuts each contributing to the probability of a randomly selected nut being in that cookie. Therefore,

$$p_K(k) = c \cdot k \cdot p_X(k)$$

Normalizing to find the constant c:

$$\sum_{k=1}^{\infty} c \cdot k \cdot p_X(k) = ce^{-1.5}\sum_{k=1}^{\infty}\frac{1.5^k}{(k-1)!} = 1$$

$$ce^{-1.5}6.72253 = 1 \quad \rightarrow \quad c = 30.1283$$

$$\boxed{p_K(k) = 30.1283\frac{1.5^k e^{-1.5}}{(k-1)!}}$$

e. Call Y the nuts per cookie after discarding all cookies with zero nuts. After discarding the cookies containing zero nuts, the relative probabilities of cookies containing greater numbers nuts are left unchanged. Therefore, we just need to scale the original Poisson distribution by a constant, and set $p_Y(0) = 0$.

$$p_Y(y) = cp_X(y), \quad y \geq 1$$

$$c\sum_{y=1}^{\infty} p_X(y) = c(1 - p_X(0)) = c(1 - e^{-1.5}) = 1 \quad \rightarrow \quad c = 1.2872$$

Computing the mean and variance:

Since the term for $X = 0$ does not contribute to the mean,

$$\boxed{\mu_Y = c\mu_X = 1.2872 \cdot 1.5 = 1.931}$$

Similarly,

$$\sigma_Y^2 = \mathbb{E}[Y^2] - \mathbb{E}[Y]^2 = c(\sigma_X^2 + \mu_X^2) - (c\mu_X)^2 = (2c - c^2)1.5^2$$

$$\boxed{\sigma_Y^2 = 2.0644}$$

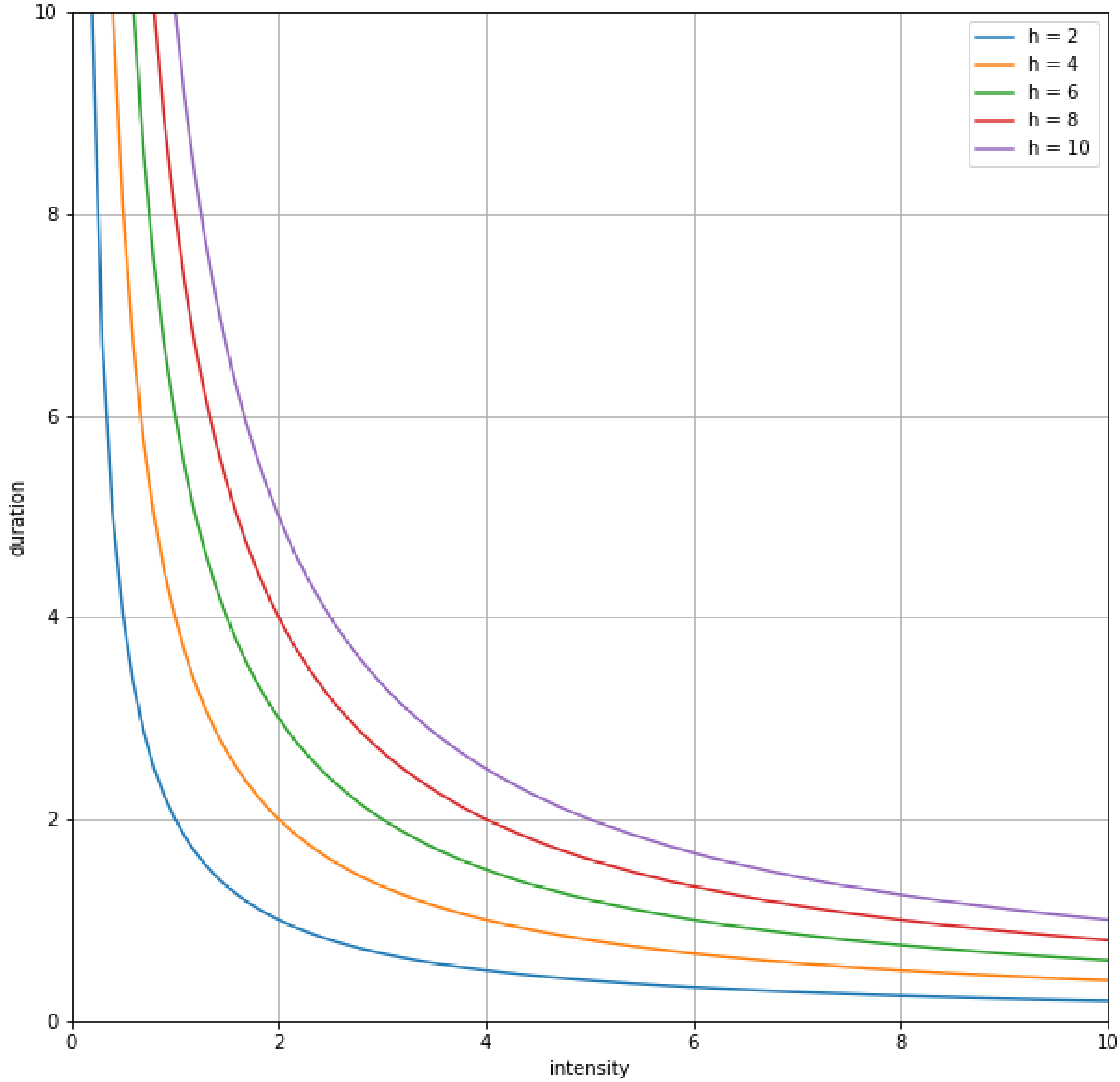
4.

a. Since they are independent, the joint distribution is just the product:

$$\boxed{f_{I,D}(i,d) = \alpha\delta e^{-\alpha i-\delta d}}$$

b. A plot of contours for constant h:

```
fig, ax = plt.subplots(1, 1, figsize=(10, 10))
ax.set_xlabel('intensity')
ax.set_ylabel('duration')
ax.set_xlim(0, 10)
ax.set_ylim(0, 10)
intensities = np.linspace(0.1, 10, 101)
depths = range(2, 11, 2)
for depth in depths:
    durations = [depth/intensity for intensity in intensities]
    ax.plot(intensities, durations, label='h = %i' %depth)
ax.legend()
ax.grid()
```



c. The region where a given depth h is not exceeded correspond to the area in the first quadrant and below the corresponding contour for h.

d.

$$F_H(h) = \int_0^\infty \int_0^{h/i} f_{I,D}(i,d) dddi = \alpha\delta \int_0^\infty e^{-\alpha i} \int_0^{h/i} e^{-\delta d} dddi$$

$$F_H(h) = \alpha \int_0^\infty e^{-\alpha i}(1 - e^{-\delta h/i}) di = 1 - \alpha \int_0^\infty e^{-\alpha i} e^{-\delta h/i} di$$

$$\boxed{F_H(h) = 1 - 2\sqrt{\alpha\delta h}\, K_1\left(2\sqrt{\alpha\delta h}\right)}$$

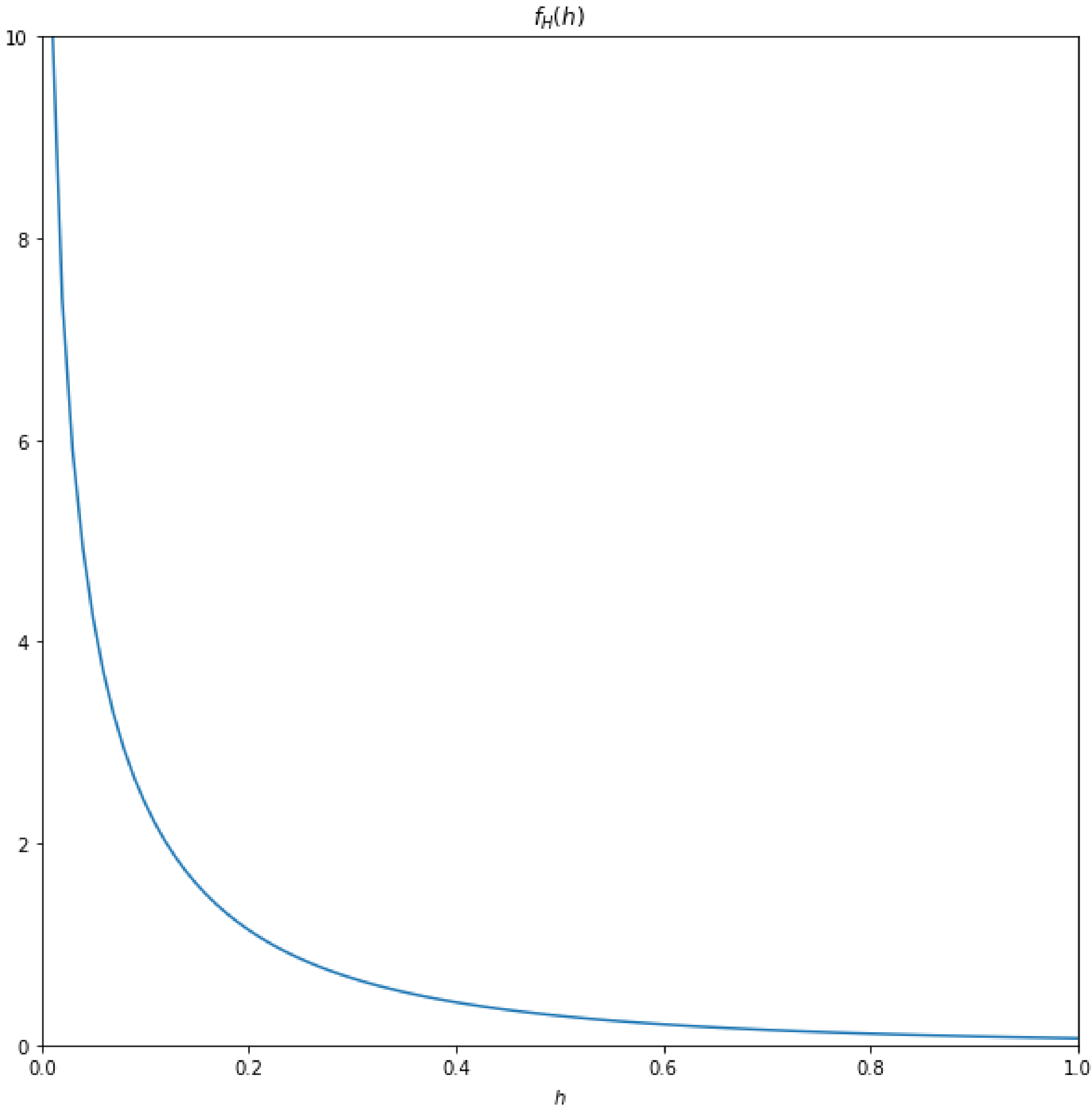
where K_1 is the modified Bessel function of the second kind.

e.

$$\boxed{f_H(h) = \frac{dF_H(h)}{dh} = 2\alpha\delta K_0\left(2\sqrt{\alpha\delta h}\right)}$$

f. Using $\alpha = 5$ and $\delta = 1$ to plot the density:

```
fig, ax = plt.subplots(1, 1, figsize=(10, 10))
xs = np.linspace(0.01, 1, 101)
y = [2*5*1*scipy.special.kn(0, 2*np.sqrt(5*1*x)) for x in xs]
ax.plot(xs, y)
ax.set_xlim(0, 1)
ax.set_ylim(0, 10)
ax.set_xlabel(r'$h$')
ax.set_title(r'$f_H(h)$')
plt.show()
```



These results look familiar, it looks like a plot of storm magnitude vs frequency.

g.

$$\boxed{\mu_H = \int_0^\infty h f_H(h) dh = \frac{1}{5} = 0.2}$$

$$\boxed{\sigma_H^2 = \int_0^\infty h^2 f_H(h) dh - \mu_H^2 = \frac{4}{25} - 0.2^2 = 0.16 - 0.04 = 0.12}$$

The results make sense because the mean depth is equal to the product of mean intensity and mean duration, as expected. The variance also obeys the property for a product of independent random variables that $Var(XY) =$

$\sigma_X^2\sigma_Y^2 + \sigma_X^2\mu_Y^2 + \sigma_Y^2\mu_X^2 = 0.04 + 0.04 + 0.04 = 0.12$

5.

a.

$$P(\text{first 4 days rainy}) = P(R_1 \cap R_2 \cap R_3 \cap R_4) = P(R_1)P(R_2|R_1)P(R_3|R_2)P(R_4|R_3)$$

$$P(R_1 \cap R_2 \cap R_3 \cap R_4) = \frac{10}{80}\cdot\frac{10}{50}\cdot\frac{10}{50}\cdot\frac{10}{50} = \boxed{0.001}$$

b. The probability that at least 3 dry days follow a given rainy day is simply:

$$P(D|R)P(D|D)P(D|D) = \frac{4}{5}\cdot\frac{9}{10}\cdot\frac{9}{10} = \boxed{\frac{81}{125} = 0.648}$$

c. The Markov chain can be represented using a state vector for the probabilities of it being rainy or dry on day n (R_n or D_n):

$$\vec{x}_n = \begin{bmatrix} P(R_n) \\ P(D_n) \end{bmatrix} \quad \rightarrow \quad \vec{x}_1 = \begin{bmatrix} \frac{1}{8} \\ \frac{7}{8} \end{bmatrix}$$

and a state transition matrix

$$A = \begin{bmatrix} P(R_{n+1}|R_n) & P(R_{n+1}|D_n) \\ P(D_{n+1}|R_n) & P(D_{n+1}|D_n) \end{bmatrix} = \begin{bmatrix} \frac{1}{5} & \frac{1}{10} \\ \frac{4}{5} & \frac{9}{10} \end{bmatrix}$$

Then the state vector evolves according to the iterative function

$$\vec{x}_{n+1} = A\vec{x}_n$$

A has two eigenvalues $\lambda_1 = 0.1, \lambda_2 = 1$ with two corresponding eigenvectors so A is diagonalizable as $A = PDP^{-1}$. Then the state vector on day n is:

$$\vec{x}_n = A^{n-1}\vec{x}_1 = PD^{n-1}P^{-1}\vec{x}_1$$

Five days after the start of the experiment, the probabilities of having a rainy or dry day are represented by the state vector

$$\vec{x}_6 = A^5\vec{x}_1 = PD^5P^{-1}\vec{x}_1$$

```
P = [[1, 1],[-1, 8]]
def D(n):
    return [[0.1**n, 0],[0, 1]]
Pinv = [[8.0/9, -1.0/9],[1.0/9, 1.0/9]]

np.matmul(np.matmul(np.matmul(P, D(5)), Pinv), [1.0/8, 7.0/8])
```

```
array([ 0.11111125,  0.88888875])
```

$$\boxed{\vec{x}_6 = \begin{bmatrix} 0.111 \\ 0.889 \end{bmatrix}}$$

The entries of this vector correspond to the probability of it being rainy or dry after 5 days have passed, respectively.

d. The steady state vector $\vec{x}_\infty$ satisfies:

$$A\vec{x}_\infty = \vec{x}_\infty$$

which corresponds to an eigenvector of A with eigenvalue 1. The elements must also sum to 1 to represent probabilities. This corresponds to the vector:

$$\boxed{\vec{x}_\infty = \begin{bmatrix} \frac{1}{9} \\ \frac{8}{9} \end{bmatrix}}$$

This answer makes sense because the steady state probability of rain is between $P(rain_t|dry_{t-1})$ and $P(rain_t|rain_{t-1})$, and closer to the former because it is more likely that it is dry the previous day. The steady state probability of a dry day is analogously sensible.

6.

a. Probability that it does not rain in a month for all combinations of $\lambda = 10, 20$ arrivals per year, $1/\beta = 0.5, 1$ days, $1/p = 2, 5$ cells per weather disturbance.

Converting units yields $\lambda = \frac{10}{12}, \frac{20}{12}$ arrivals per month, $1/\beta = 0.5 \cdot \frac{12}{365}, \frac{12}{365}$ months.

$$P(N[0,t] = 0) = \left(\frac{p}{1 - (1-p)e^{-\beta t}}\right)^{\lambda/\beta} e^{-\lambda t}$$

Plugging in $t = 1$ month for each combination of parameters yields the following probabilities:

```
lambdas = [10.0/12, 20.0/12]
betas = [365/6.0, 365/12.0]
ps = [1.0/2, 1.0/5]

# set LaTeX table formatting for pdf export
def _repr_latex_(self):
    lself = self
    lself = lself.rename(columns=dict(zip(self.columns.tolist(), self.columns.tolist())))
    return r'\begin{center}' + '%s' % lself.to_latex(index=False, escape=False) +
     ↪  r'\end{center}'

pd.DataFrame._repr_latex_ = _repr_latex_

df = pd.DataFrame(columns=[r'$\lambda$', r'$\beta$', r'$p$', r'$P(\text{no rain})$'])

for lamb in lambdas:
    for beta in betas:
        for p in ps:
            no_rain = ((p)/(1 - (1 - p) * np.exp(-beta*1)))**(lamb/beta) * np.exp(-lamb*1)
```

```python
            row = pd.DataFrame([[lamb, beta, p, no_rain]], columns=df.columns.tolist())
            df = df.append(row, ignore_index=True)

df = df.round(6)
df
```

λ	β	p	P(no rain)
0.833333	60.833333	0.5	0.430491
0.833333	60.833333	0.2	0.425121
0.833333	30.416667	0.5	0.426423
0.833333	30.416667	0.2	0.415851
1.666667	60.833333	0.5	0.185323
1.666667	60.833333	0.2	0.180728
1.666667	30.416667	0.5	0.181837
1.666667	30.416667	0.2	0.172932

The results make sense because when p is smaller (more cells in weather disturbance), the probability of no rain in the month goes down. As λ goes up (more arrivals per month), the probability of no rain in the month goes down. And as β goes down (shorter time between cells), the probability of no rain in the month goes down.

b. Find the pdf of the time T between arrivals.

The CDF of the time between arrivals T is:

$$F_T(t) = P(T \leq t) = 1 - P(T > t) = 1 - P(N[0,t] = 0) = 1 - \left(\frac{p}{1-(1-p)e^{-\beta t}}\right)^{\lambda/\beta} e^{-\lambda t}$$

Therefore the pdf of T is:

$$\boxed{f_T(t) = \frac{dF_T(t)}{dt} = \lambda e^{-\lambda t}\left(\frac{p}{1-(1-p)e^{-\beta t}}\right)^{\lambda/\beta}\left[1 - \frac{(1-p)e^{-\beta t}}{1-(1-p)e^{-\beta t}}\right]}$$

Again, this makes sense because each term in the pdf decreases as t increases. The leading term is the pdf for an exponential distribution with parameter λ, so this distribution drops off slightly faster than the exponential due to the additional clustering of children.