

KL Divergence and Transformations of Random Variables

Matthew Seguin

1.

We are considering 3 approximations to the binomial.

Take $Y \sim \text{Poisson}(np)$ then the first approximation is:

$$\binom{n}{k} p^k (1-p)^{n-k} = \mathbb{P}[X = k] \approx \mathbb{P}[Y = k] = \frac{e^{-np} (np)^k}{k!}$$

Using the normal density function:

$$\binom{n}{k} p^k (1-p)^{n-k} = \mathbb{P}[X = k] \approx \phi\left(k; np, np(1-p)\right) = \frac{1}{\sqrt{2n\pi p(1-p)}} e^{-\frac{(k-np)^2}{2np(1-p)}}$$

The entropic approximation (where $f = \frac{k}{n}$ and $\text{KL}(f||p) = f \log \frac{f}{p} + (1-f) \log \frac{1-f}{1-p}$):

$$\binom{n}{k} p^k (1-p)^{n-k} = \mathbb{P}[X = k] \approx \text{Ent}(k; n, p) = \frac{1}{\sqrt{2n\pi f(1-f)}} e^{-n \text{KL}(f||p)}$$

The absolute errors are calculated directly from the formulas:

$$\text{Poisson Approx. Absolute Error: } \left| \binom{n}{k} p^k (1-p)^{n-k} - \frac{e^{-np} (np)^k}{k!} \right|$$

$$\text{Normal Approx. Absolute Error: } \left| \binom{n}{k} p^k (1-p)^{n-k} - \frac{1}{\sqrt{2n\pi p(1-p)}} e^{-\frac{(k-np)^2}{2np(1-p)}} \right|$$

$$\text{Entropic Approx. Absolute Error: } \left| \binom{n}{k} p^k (1-p)^{n-k} - \frac{1}{\sqrt{2n\pi f(1-f)}} e^{-n \text{KL}(f||p)} \right|$$

Similarly the relative errors are calculated directly from the formulas:

$$\text{Poisson Approx. Relative Error: } \frac{\left| \binom{n}{k} p^k (1-p)^{n-k} - \frac{e^{-np} (np)^k}{k!} \right|}{\binom{n}{k} p^k (1-p)^{n-k}}$$

$$\text{Normal Approx. Relative Error: } \frac{\left| \binom{n}{k} p^k (1-p)^{n-k} - \frac{1}{\sqrt{2n\pi p(1-p)}} e^{-\frac{(k-np)^2}{2np(1-p)}} \right|}{\binom{n}{k} p^k (1-p)^{n-k}}$$

$$\text{Entropic Approx. Relative Error: } \frac{\left| \binom{n}{k} p^k (1-p)^{n-k} - \frac{1}{\sqrt{2n\pi f(1-f)}} e^{-n \text{KL}(f||p)} \right|}{\binom{n}{k} p^k (1-p)^{n-k}}$$

a.

First we are using $n = 30$ and $p = 0.05$.

From the code shown [here](#) we got the absolute errors below:

k	Poisson Absolute Errors	Normal Absolute Errors	Entropic Absolute Errors
0	8.491396e-03	6.288532e-02	
1	4.208071e-03	3.277279e-02	2.865137e-02
2	7.615308e-03	4.749378e-02	1.096979e-02
3	1.538910e-03	2.470382e-02	3.605901e-03
4	1.930467e-03	7.845186e-03	9.679277e-04
5	1.766931e-03	7.810490e-03	2.143120e-04
6	8.209923e-04	2.434696e-03	3.976033e-05
7	2.675847e-04	4.806307e-04	6.268669e-06
8	6.786047e-05	7.384760e-05	8.496632e-07
9	1.412178e-05	9.515639e-06	9.992828e-08
10	2.493920e-06	1.051824e-06	1.027277e-08
11	3.828577e-07	1.006534e-07	9.284106e-10
12	5.205110e-08	8.387780e-09	7.409012e-11
13	6.362462e-09	6.112552e-10	5.237982e-12
14	7.081188e-10	3.906518e-11	3.287950e-13
15	7.252526e-11	2.193133e-12	1.834906e-14
16	6.896636e-12	1.082138e-13	9.107894e-16
17	6.133846e-13	4.690382e-15	4.019292e-17
18	5.132796e-14	1.782894e-16	1.574848e-18
19	4.060356e-15	5.926516e-18	5.466524e-20
20	3.047997e-16	1.715570e-19	1.675528e-21
21	2.177936e-17	4.299675e-21	4.514869e-23
22	1.485157e-18	9.257674e-23	1.063399e-24
23	9.686240e-20	1.694769e-24	2.173290e-26
24	6.053980e-21	2.601619e-26	3.818433e-28
25	3.632400e-22	3.286255e-28	5.701307e-30
26	2.095617e-23	3.326169e-30	7.132860e-32
27	1.164232e-24	2.593504e-32	7.360840e-34
28	6.236956e-26	1.462502e-34	6.203044e-36
29	3.226012e-27	5.308539e-37	4.487914e-38
30	1.613006e-28	9.313226e-40	

As we can see the absolute errors are decreasing as k increases, all of them seem to perform relatively well with the Poisson approximation seeming to be best at lower values of k and the Entropic approximation seeming to do best overall.

b.

Again we are using $n = 30$ and $p = 0.05$.

From the code shown [here](#) we got the absolute errors below:

k	Poisson Relative Errors	Normal Relative Errors	Entropic Relative Errors
0	3.956134e-02	0.29298210	
1	1.241673e-02	0.09670249	0.084541418
2	2.944403e-02	0.18363122	0.042413904
3	1.211267e-02	0.19444231	0.028381831
4	4.276996e-02	0.17381198	0.021444669
5	1.430363e-01	0.63227346	0.017348947
6	3.030614e-01	0.89874474	0.014677142
7	5.473854e-01	0.98320354	0.012823520
8	9.174123e-01	0.99835288	0.011486679
9	1.483921e+00	0.99990583	0.010500489
10	2.371035e+00	0.99999689	0.009766593
11	3.803725e+00	0.99999994	0.009223841
12	6.205587e+00	1.00000000	0.008833103
13	1.040885e+01	1.00000000	0.008569223
14	1.812660e+01	1.00000000	0.008416573
15	3.306925e+01	1.00000000	0.008366598
16	6.373157e+01	1.00000000	0.008416573
17	1.307750e+02	1.00000000	0.008569223
18	2.878913e+02	1.00000000	0.008833103
19	6.851169e+02	1.00000000	0.009223841
20	1.776666e+03	1.00000000	0.009766593
21	5.065349e+03	1.00000000	0.010500489
22	1.604244e+04	1.00000000	0.011486679
23	5.715375e+04	1.00000000	0.012823520
24	2.327005e+05	1.00000000	0.014677142
25	1.105331e+06	1.00000000	0.017348947
26	6.300392e+06	1.00000000	0.021444669
27	4.489030e+07	1.00000000	0.028381831
28	4.264579e+08	1.00000000	0.042413904
29	6.077024e+09	1.00000000	0.084541418
30	1.731952e+11	1.00000000	

As we can see the relative errors are increasing as k increases. This is different than what we saw before. The reason is that although the absolute errors were decreasing, the probability was decreasing faster. So the relative error actually increases with k . They don't seem to perform that well according to this as for some of the higher k values the absolute error is equal to, or even greater than, the probability itself. Here we see the same result that Poisson approximation seeming to be best at lower values of k and the Entropic approximation seeming to do best overall.

c.

Now we are using $n = 30$ and $p = 0.25$.

From the code shown here we got the errors below:

First here are the absolute errors:

k	Poisson Absolute Errors	Normal Absolute Errors	Entropic Absolute Errors
0	3.745023e-04	9.548001e-04	
1	2.362312e-03	2.148155e-03	1.509758e-04
2	6.924030e-03	2.799400e-03	3.660942e-04
3	1.203529e-02	9.512783e-04	7.621502e-04
4	1.249612e-02	3.802816e-03	1.295693e-03
5	4.646120e-03	8.218080e-03	1.816929e-03
6	8.737971e-03	7.738469e-03	2.134882e-03
7	1.975184e-02	1.723586e-03	2.131727e-03
8	2.198059e-02	5.202900e-03	1.829934e-03
9	1.536699e-02	7.910260e-03	1.363042e-03
10	5.034870e-03	5.645154e-03	8.874438e-04
11	3.450864e-03	1.547615e-03	5.079555e-04
12	7.510803e-03	1.259906e-03	2.567309e-04
13	7.686768e-03	1.983581e-03	1.149514e-04
14	5.874565e-03	1.495681e-03	4.569911e-05
15	3.721567e-03	7.971628e-04	1.615209e-05
16	2.046132e-03	3.299491e-04	5.077679e-06
17	1.003255e-03	1.104228e-04	1.419153e-06
18	4.471579e-04	3.054174e-05	3.521686e-07
19	1.838540e-04	7.073812e-06	7.742043e-08
20	7.055401e-05	1.382503e-06	1.502894e-08
21	2.550318e-05	2.287576e-07	2.564804e-09
22	8.744228e-06	3.202122e-08	3.825936e-10
23	2.858378e-06	3.772368e-09	4.952123e-11
24	8.940745e-07	3.702489e-10	5.510502e-12
25	2.683049e-07	2.978268e-11	5.210901e-13
26	7.740240e-08	1.915055e-12	4.128901e-14
27	2.150111e-08	9.472807e-14	2.698550e-15
28	5.759247e-09	3.385663e-15	1.440258e-16
29	1.489461e-09	7.782203e-17	6.599519e-18
30	3.723653e-10	8.625467e-19	

Now it is just clear that the Entropic approximation is the best overall having the lowest errors overall. Although, all of the approximations seem to be accurate. As before the absolute errors are generally decreasing with k .

Continued on next page.

Now here are the relative errors:

k	Poisson Relative Errors	Normal Relative Errors	Entropic Relative Errors
0	2.097088e+00	5.34656148	
1	1.322816e+00	1.20289525	0.084541418
2	8.021846e-01	0.32432490	0.042413904
3	4.481840e-01	0.03542480	0.028381831
4	2.068200e-01	0.06293940	0.021444669
5	4.436348e-02	0.07847035	0.017348947
6	6.007286e-02	0.05320136	0.014677142
7	1.188183e-01	0.01036833	0.012823520
8	1.379744e-01	0.03265913	0.011486679
9	1.183829e-01	0.06093839	0.010500489
10	5.541030e-02	0.06212666	0.009766593
11	6.266342e-02	0.02810277	0.009223841
12	2.584172e-01	0.04334841	0.008833103
13	5.730215e-01	0.14786901	0.008569223
14	1.081940e+00	0.27546514	0.008416573
15	1.927728e+00	0.41292112	0.008366598
16	3.391593e+00	0.54691136	0.008416573
17	6.057917e+00	0.66676229	0.008569223
18	1.121563e+01	0.76604869	0.008833103
19	2.190430e+01	0.84277120	0.009223841
20	4.584970e+01	0.89842291	0.009766593
21	1.044118e+02	0.93654972	0.010500489
22	2.625296e+02	0.96137894	0.011486679
23	7.401769e+02	0.97685460	0.012823520
24	2.381354e+03	0.98615246	0.014677142
25	8.932828e+03	0.99157154	0.017348947
26	4.020123e+04	0.99464042	0.021444669
27	2.261365e+05	0.99629656	0.028381831
28	1.696030e+06	0.99703786	0.042413904
29	1.908035e+07	0.99691874	0.084541418
30	4.293080e+08	0.99444867	

It is still clear that the Entropic approximation is the best overall having the lowest errors overall. The Poisson approximation seems to do rather poorly when looking at relative errors, normal doesn't look too good either. As before the relative errors are generally increasing with k . The reason is that although the absolute errors were decreasing, the probability was decreasing faster. So the relative error actually increases with k .

d.

Now we are using $n = 30$ and $p = 0.5$.

From the code shown here we got the errors below:

First here are the absolute errors:

k	Poisson Absolute Errors	Normal Absolute Errors	Entropic Absolute Errors
0	3.049710e-07	4.363042e-08	
1	4.560595e-06	2.800940e-07	2.362060e-09
2	3.400889e-05	1.458370e-06	1.718295e-08
3	1.682889e-04	6.085087e-06	1.073165e-07
4	6.197398e-04	2.019336e-05	5.473300e-07
5	1.803069e-03	5.266949e-05	2.302536e-06
6	4.286474e-03	1.049484e-04	8.116402e-06
7	8.474307e-03	1.475841e-04	2.431322e-05
8	1.399334e-02	1.040396e-04	6.261344e-05
9	1.908260e-02	1.094041e-04	1.399145e-04
10	2.062915e-02	4.675016e-04	2.732849e-04
11	1.541175e-02	7.416804e-04	4.692688e-04
12	2.306141e-03	6.059874e-04	7.115337e-04
13	1.592824e-02	4.012203e-05	9.557687e-04
14	3.299955e-02	8.428050e-04	1.139902e-03
15	4.202858e-02	1.208676e-03	1.208676e-03
16	3.940180e-02	8.428050e-04	1.139902e-03
17	2.679950e-02	4.012203e-05	9.557687e-04
18	9.940133e-03	6.059874e-04	7.115337e-04
19	4.871436e-03	7.416804e-04	4.692688e-04
20	1.382870e-02	4.675016e-04	2.732849e-04
21	1.653993e-02	1.094041e-04	1.399145e-04
22	1.491120e-02	1.040396e-04	6.261344e-05
23	1.138368e-02	1.475841e-04	2.431322e-05
24	7.746798e-03	1.049484e-04	8.116402e-06
25	4.847157e-03	5.266949e-05	2.302536e-06
26	2.847483e-03	2.019336e-05	5.473300e-07
27	1.592333e-03	6.085087e-06	1.073165e-07
28	8.546561e-04	1.458370e-06	1.718295e-08
29	4.422451e-04	2.800940e-07	2.362060e-09
30	2.211356e-04	4.363042e-08	

Now we see that the normal approximation performs the best in the middle values of k but I still think the Entropic approximation is the best overall having the lowest errors overall. Once again, all of the approximations seem to be accurate (though I suspect this will be a different case when we look at the relative errors). Now the relative errors are generally increasing as k moves away from the middle.

Continued on next page.

Now here are the relative errors:

k	Poisson Relative Errors	Normal Relative Errors	Entropic Relative Errors
0	3.274601e+02	4.684781e+01	
1	1.632301e+02	1.002495e+01	0.084541418
2	8.394658e+01	3.599799e+00	0.042413904
3	4.450710e+01	1.609313e+00	0.028381831
4	2.428172e+01	7.911860e-01	0.021444669
5	1.358561e+01	3.968495e-01	0.017348947
6	7.751365e+00	1.897814e-01	0.014677142
7	4.469603e+00	7.784027e-02	0.012823520
8	2.567132e+00	1.908647e-02	0.011486679
9	1.432136e+00	8.210703e-03	0.010500489
10	7.372398e-01	1.670746e-02	0.009766593
11	3.029299e-01	1.457830e-02	0.009223841
12	2.862884e-02	7.522833e-03	0.008833103
13	1.428093e-01	3.597258e-04	0.008569223
14	2.436553e-01	6.222929e-03	0.008416573
15	2.909268e-01	8.366598e-03	0.008366598
16	2.909268e-01	6.222929e-03	0.008416573
17	2.402787e-01	3.597258e-04	0.008569223
18	1.233985e-01	7.522833e-03	0.008833103
19	9.575183e-02	1.457830e-02	0.009223841
20	4.942070e-01	1.670746e-02	0.009766593
21	1.241311e+00	8.210703e-03	0.010500489
22	2.735518e+00	1.908647e-02	0.011486679
23	6.004096e+00	7.784027e-02	0.012823520
24	1.400878e+01	1.897814e-01	0.014677142
25	3.652194e+01	3.968495e-01	0.017348947
26	1.115658e+02	7.911860e-01	0.021444669
27	4.211218e+02	1.609313e+00	0.028381831
28	2.109609e+03	3.599799e+00	0.042413904
29	1.582857e+04	1.002495e+01	0.084541418
30	2.374425e+05	4.684781e+01	

It is still clear that the Entropic approximation is the best overall having the lowest errors overall. The Poisson approximation seems to do rather poorly when looking at relative errors (only good at the middle values of k), normal is actually the best for only the very middle values of k but otherwise isn't very good. Now the relative errors are generally increasing as k moves away from the middle.

2.

Let X and Y be discrete with $p(k) = \mathbb{P}[X = k]$ and $q(k) = \mathbb{P}[Y = k]$ respectively.

a.

Assume that $Y \sim \text{Poisson}(\lambda)$ for some $\lambda > 0$, that is $q(k) = \frac{e^{-\lambda} \lambda^k}{k!}$ for some $\lambda > 0$.

Then for the KL divergence we have:

$$\text{KL}(p||q) = \sum_{k=0}^{\infty} p(k) \log \frac{p(k)}{q(k)} = \sum_{k=0}^{\infty} p(k) \log \left(p(k) \frac{k!}{e^{-\lambda} \lambda^k} \right) = \sum_{k=0}^{\infty} p(k) \left(\lambda + \log(p(k)) + \log(k!) - k \log(\lambda) \right)$$

To minimize this with respect to λ we can take the derivative with respect to λ :

$$\begin{aligned} \frac{d}{d\lambda} \text{KL}(p||q) &= \frac{d}{d\lambda} \sum_{k=0}^{\infty} p(k) \left(\lambda + \log(p(k)) + \log(k!) - k \log(\lambda) \right) \\ &= \sum_{k=0}^{\infty} \frac{d}{d\lambda} p(k) \left(\lambda + \log(p(k)) + \log(k!) - k \log(\lambda) \right) = \sum_{k=0}^{\infty} p(k) \left(1 - \frac{k}{\lambda} \right) \\ &= \left(\sum_{k=0}^{\infty} p(k) \right) - \left(\frac{1}{\lambda} \sum_{k=0}^{\infty} k p(k) \right) = 1 - \frac{1}{\lambda} \mathbb{E}[X] \end{aligned}$$

Then setting this equal to 0 we have:

$$1 - \frac{1}{\lambda} \mathbb{E}[X] = 0 \implies \lambda = \mathbb{E}[X]$$

So $\mathbb{E}[X]$, which is just the mean of $p(k)$, is our critical point.

Then checking that this is indeed a minimum we can take the second derivative:

$$\frac{d^2}{d\lambda^2} \text{KL}(p||q) = \frac{d}{d\lambda} 1 - \frac{1}{\lambda} \mathbb{E}[X] = \frac{1}{\lambda^2} \mathbb{E}[X] > 0$$

Since X is a non-negative random variable, and by assumption we say $p(0) = \mathbb{P}[X = 0] < 1$.

Therefore $\text{KL}(p||q)$ is concave up with respect to λ and hence $\text{KL}(p||q)$ is minimized our critical point $\lambda = \mathbb{E}[X]$ \square

b.

Imagine placing n balls into d bins, the number of ways to place k_i balls in bin i is $\binom{n}{k_1, \dots, k_d}$

Let N_p denote the number of configurations with empirical distribution $p(i) = \frac{k_i}{n}$.

Since p is entirely determined by the number of balls in each box we get that:

$$\begin{aligned} N_p &= \binom{n}{k_1, \dots, k_d} = \left(\frac{n!}{k_1!(n-k_1)!} \right) \left(\frac{(n-k_1)!}{k_2!(n-k_1-k_2)!} \right) \dots \left(\frac{(n-k_1-\dots-k_{d-2})!}{k_{d-1}!(n-k_1-\dots-k_{d-1})!} \right) \left(\frac{(n-k_1-\dots-k_{d-1})!}{k_d!(n-k_1-\dots-k_d)!} \right) \\ &= \left(\frac{n!}{k_1!} \right) \left(\frac{1}{k_2!(n-k_1-k_2)!} \right) \left(\frac{(n-k_1-k_2)!}{k_3!(n-k_1-k_2-k_3)!} \right) \dots \left(\frac{(n-k_1-\dots-k_{d-2})!}{k_{d-1}!(n-k_1-\dots-k_{d-1})!} \right) \left(\frac{(n-k_1-\dots-k_{d-1})!}{k_d!0!} \right) \\ &= \dots = \frac{n!}{k_1!k_2!\dots k_d!} \end{aligned}$$

Since each subsequent term cancels out the $n-k_1-\dots-k_j$ term from the previous and $n-k_1-\dots-k_d=0$ so the final term just has a $0!=1$ in the denominator. Therefore:

$$\begin{aligned} \log N_p &= \log \frac{n!}{k_1!\dots k_d!} = \log n! - \sum_{i=1}^d \log k_i! \\ &= n \log n - n + O(\log(n)) - \sum_{i=1}^d k_i \log k_i - k_i + O(\log(k_i)) \\ &= n \log n - n - \left(\sum_{i=1}^d k_i \log k_i \right) + \left(\sum_{i=1}^d k_i \right) + O(\log(n)) - \sum_{i=1}^d O(\log(k_i)) \\ &= n \log n - \left(\sum_{i=1}^d k_i \log k_i \right) + O(\log(n)) - \sum_{i=1}^d O(\log(k_i)) \\ &= n \log n - \left(\sum_{i=1}^d k_i \log k_i \right) + O(\log(n)) \end{aligned}$$

$O(\log(n))$ is the dominating convergence term since $\log k_i \leq \log n$ for each i ($k_i \leq n$).

Now we consider $H(p)$:

$$\begin{aligned} H(p) &= - \sum_{i=1}^d \frac{k_i}{n} \log \frac{k_i}{n} = - \frac{1}{n} \sum_{i=1}^d k_i (\log k_i - \log n) = \frac{1}{n} \sum_{i=1}^d k_i (\log n - \log k_i) \\ &= \left(\frac{1}{n} \sum_{i=1}^d k_i \log n \right) - \left(\frac{1}{n} \sum_{i=1}^d k_i \log k_i \right) = \left(\log n \sum_{i=1}^d \frac{k_i}{n} \right) - \left(\frac{1}{n} \sum_{i=1}^d k_i \log k_i \right) = \log n - \left(\frac{1}{n} \sum_{i=1}^d k_i \log k_i \right) \end{aligned}$$

So we have that: $nH(p) = n \log n - \left(\sum_{i=1}^d k_i \log k_i \right)$

Therefore:

$$\log N_p = n \log n - \left(\sum_{i=1}^d k_i \log k_i \right) + O(\log(n)) = nH(p) + O(\log(n)) \quad \square$$

3.

Let $N \sim \text{Poisson}(\lambda)$ then let $X_1, \dots, X_n \stackrel{\text{iid}}{\sim} \text{Bernoulli}(p)$ and $K = X_1 + \dots + X_N$.

We will further assume N and $\{X_i : i \in \mathbb{N}\}$ are mutually independent.

Clearly $K|N \sim \text{Binomial}(N, p)$ so $\mathbb{P}[K = k|N = n] = \binom{n}{k} p^k (1-p)^{n-k}$ for $0 \leq k \leq n$.

We also know that $\mathbb{P}[N = n] = \frac{e^{-\lambda} \lambda^n}{n!}$ for $0 \leq k < \infty$, therefore:

$$\begin{aligned} \mathbb{P}[K = k] &= \sum_{n=0}^{\infty} \mathbb{P}[N = n] \mathbb{P}[K = k|N = n] = \sum_{n=k}^{\infty} \frac{e^{-\lambda} \lambda^n}{n!} \binom{n}{k} p^k (1-p)^{n-k} \\ &= \sum_{n=k}^{\infty} \frac{e^{-\lambda} \lambda^n}{n!} \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k} = \frac{e^{-\lambda} (\lambda p)^k}{k!} \sum_{n=k}^{\infty} \frac{\lambda^{n-k}}{(n-k)!} (1-p)^{n-k} \\ &= \frac{e^{-\lambda} (\lambda p)^k}{k!} \sum_{j=0}^{\infty} \frac{(\lambda(1-p))^j}{(j)!} = \frac{e^{-\lambda} (\lambda p)^k}{k!} e^{\lambda(1-p)} = \frac{e^{-\lambda p} (\lambda p)^k}{k!} \end{aligned}$$

Which we recognize as $\mathbb{P}[Y = k]$ for $Y \sim \text{Poisson}(\lambda p)$.

Therefore: $K \sim \text{Poisson}(\lambda p)$ \square

4.

We consider the joint random variables X and Y with joint density:

$$f_{X,Y}(x,y) = \begin{cases} 3xy(x+y) & \text{for } (x,y) \in [0,1]^2 \\ 0 & \text{otherwise} \end{cases}$$

First let us just verify this is a joint density. Clearly $f_{X,Y}(x,y) \geq 0$ for all $(x,y) \in \mathbb{R}^2$ then:

$$\begin{aligned} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{X,Y}(x,y) dx dy &= \int_0^1 \int_0^1 3xy(x+y) dx dy = \int_0^1 \int_0^1 3x^2y + 3xy^2 dx dy \\ &= \int_0^1 \left(x^3y + \frac{3}{2}x^2y^2 \Big|_0^1 \right) dy = \int_0^1 y + \frac{3}{2}y^2 dy = \frac{1}{2}y^2 + \frac{1}{2}y^3 \Big|_0^1 = \frac{1}{2} + \frac{1}{2} = 1 \end{aligned}$$

So we know that $f_{X,Y}(x,y)$ is indeed a joint density.

Now we will find the marginal densities and expectations of X and Y :

$$\begin{aligned} f_X(x) &= \int_{-\infty}^{\infty} f_{X,Y}(x,y) dy = \int_0^1 3xy(x+y) dy = \int_0^1 3x^2y + 3xy^2 dy \\ &= \frac{3}{2}x^2y^2 + xy^3 \Big|_0^1 = \frac{3}{2}x^2 + x \end{aligned}$$

$$\mathbb{E}[X] = \int_{-\infty}^{\infty} x f_X(x) dx = \int_0^1 x \left(\frac{3}{2}x^2 + x \right) dx = \int_0^1 \frac{3}{2}x^3 + x^2 dx = \frac{3}{8}x^4 + \frac{1}{3}x^3 \Big|_0^1 = \frac{3}{8} + \frac{1}{3} = \frac{17}{24}$$

$$\begin{aligned} f_Y(y) &= \int_{-\infty}^{\infty} f_{X,Y}(x,y) dx = \int_0^1 3xy(x+y) dx = \int_0^1 3x^2y + 3xy^2 dx \\ &= x^3y + \frac{3}{2}x^2y^2 \Big|_0^1 = y + \frac{3}{2}y^2 \end{aligned}$$

$$\mathbb{E}[Y] = \int_{-\infty}^{\infty} y f_Y(y) dy = \int_0^1 y \left(y + \frac{3}{2}y^2 \right) dy = \int_0^1 \frac{3}{2}y^3 + y^2 dy = \frac{3}{8}y^4 + \frac{1}{3}y^3 \Big|_0^1 = \frac{3}{8} + \frac{1}{3} = \frac{17}{24}$$

We could have also noticed that $f_{X,Y}(x,y) = 3x^2y + 3xy^2 = 3y^2x + 3yx^2 = f_{Y,X}(y,x)$ so the joint density is symmetric in X and Y and hence the marginal distributions of X and Y are the same.

Now we will find $\mathbb{E}[XY]$:

$$\begin{aligned} \mathbb{E}[XY] &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} xy f_{X,Y}(x,y) dx dy = \int_0^1 \int_0^1 xy (3xy(x+y)) dx dy \\ &= \int_0^1 \int_0^1 3x^3y^2 + 3x^2y^3 dx dy = \int_0^1 \left(\frac{3}{4}x^4y^2 + x^3y^3 \Big|_0^1 \right) dy = \int_0^1 \frac{3}{4}y^2 + y^3 dy \\ &= \frac{1}{4}y^3 + \frac{1}{4}y^4 \Big|_0^1 = \frac{1}{4} + \frac{1}{4} = \frac{1}{2} \end{aligned}$$

$$\text{Therefore } \text{Cov}(X,Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y] = \frac{1}{2} - \left(\frac{17}{24} \right)^2 = \frac{288-289}{576} = -\frac{1}{576}$$

5.

a.

Let X have the Cauchy distribution, that is $f_X(x) = \frac{1}{\pi(1+x^2)}$ for $-\infty < x < \infty$

Now let $Y = g(X) = \frac{1}{X}$.

Let $y > 0$, then:

$$1 - F_Y(y) = \mathbb{P}[Y > y] = \mathbb{P}\left[\frac{1}{X} > y\right] = \mathbb{P}\left[0 < X < \frac{1}{y}\right] = 1 - \mathbb{P}[(X \leq 0) \cup (X \geq \frac{1}{y})]$$

$$= 1 - \mathbb{P}[X \leq 0] - \mathbb{P}[X \geq \frac{1}{y}] = \mathbb{P}[X < \frac{1}{y}] - \mathbb{P}[X \leq 0] = F_X(\frac{1}{y}) - F_X(0)$$

So $F_Y(y) = 1 - F_X(\frac{1}{y}) + F_X(0)$ and therefore:

$$f_Y(y) = \frac{d}{dy} F_Y(y) = \frac{d}{dy} \left(1 - F_X(\frac{1}{y}) + F_X(0) \right) = \frac{1}{y^2} f_X(\frac{1}{y}) = \frac{1}{y^2 (\pi(1 + (\frac{1}{y})^2))} = \frac{1}{\pi(1 + y^2)}$$

This was only for $y > 0$ though, now we will do the same for $y < 0$ (note this is going to be very similar since X is symmetric and so Y will be):

$$F_Y(y) = \mathbb{P}[Y \leq y] = \mathbb{P}\left[\frac{1}{X} \leq y\right] = \mathbb{P}\left[\frac{1}{y} \leq X < 0\right] = 1 - \mathbb{P}[(X \geq 0) \cup (X < \frac{1}{y})]$$

$$= 1 - \mathbb{P}[X \geq 0] - \mathbb{P}[X < \frac{1}{y}] = \mathbb{P}[X < 0] - \mathbb{P}[X < \frac{1}{y}] = F_X(0) - F_X(\frac{1}{y})$$

So $F_Y(y) = F_X(0) - F_X(\frac{1}{y})$ and therefore:

$$f_Y(y) = \frac{d}{dy} F_Y(y) = \frac{d}{dy} \left(F_X(0) - F_X(\frac{1}{y}) \right) = \frac{1}{y^2} f_X(\frac{1}{y}) = \frac{1}{y^2 (\pi(1 + (\frac{1}{y})^2))} = \frac{1}{\pi(1 + y^2)}$$

Therefore we have shown $f_Y(y) = f_X(y)$ for all $|y| > 0$, so we know:

For all $y \in \mathbb{R}$

$$F_Y(y) = \mathbb{P}[Y \leq y] = \int_{-\infty}^0 f_Y(y) dy + \int_0^y f_Y(y) dy = \int_{-\infty}^0 f_X(y) dy + \int_0^y f_X(y) dy = \int_{-\infty}^y f_X(y) dy = F_X(y)$$

Hence $Y = \frac{1}{X}$ also follows a Cauchy distribution \square

b.

First from noticing the $\frac{1}{1+x^2}$ in the Cauchy density (which is the derivative of $\tan^{-1}(x)$) we can suspect we might be doing something with \tan or \tan^{-1} . However, the Cauchy distribution is symmetric while the Exponential is not so we might first want to transform it into something that is.

The CDF of an Exponential(1) random variable is $F_X(x) = 1 - e^{-x}$ (and has an inverse CDF: $F^{-1}(x) = -\log(1 - x)$).

Then if $X \sim \text{Exponential}(1)$ we know $F(X) = 1 - e^{-X} \sim \text{Uniform}(0, 1)$. This is because of the below:

Assume Y has CDF $G(y)$ that is invertible with inverse G^{-1} and let $Z = G(Y)$, then:

$$\mathbb{P}[Z \leq z] = \mathbb{P}[G(Y) \leq z] = \mathbb{P}[Y \leq G^{-1}(z)] = G(G^{-1}(z)) = \begin{cases} 0 & z < 0 \\ z & 0 \leq z \leq 1 \\ 1 & z > 1 \end{cases}$$

Showing that $Z = G(Y)$ is $\text{Uniform}(0, 1)$.

Now we know the range of \tan^{-1} is $(-\frac{\pi}{2}, \frac{\pi}{2})$ so we want to transform our standard uniform to be $\text{Uniform}(-\frac{\pi}{2}, \frac{\pi}{2})$.

Let $U \sim \text{Uniform}(0, 1)$ and $Q = \pi U - \frac{\pi}{2}$, then:

$$\mathbb{P}[Q \leq x] = \mathbb{P}[\pi U - \frac{\pi}{2} \leq x] = \mathbb{P}[U \leq \frac{x}{\pi} + \frac{1}{2}] = \begin{cases} 0 & \frac{x}{\pi} + \frac{1}{2} < 0 \\ \frac{x}{\pi} + \frac{1}{2} & 0 \leq \frac{x}{\pi} + \frac{1}{2} \leq 1 \\ 1 & \frac{x}{\pi} + \frac{1}{2} > 1 \end{cases} = \begin{cases} 0 & x < -\frac{\pi}{2} \\ \frac{x - \frac{\pi}{2}}{\pi} & -\frac{\pi}{2} \leq x \leq \frac{\pi}{2} \\ 1 & x > \frac{\pi}{2} \end{cases}$$

Which is clearly the CDF of a $\text{Uniform}(-\frac{\pi}{2}, \frac{\pi}{2})$ random variable.

Therefore if $X \sim \text{Exponential}(1)$ we know $-\pi(1 - e^{-X}) - \frac{\pi}{2} \sim \text{Uniform}(-\frac{\pi}{2}, \frac{\pi}{2})$.

Now assume $W \sim \text{Uniform}(-\frac{\pi}{2}, \frac{\pi}{2})$ and let $C = \tan(W)$, then:

$$\mathbb{P}[C \leq c] = \mathbb{P}[\tan(W) \leq c] = \mathbb{P}[W \leq \tan^{-1}(c)] = \begin{cases} 0 & \tan^{-1}(c) \leq -\frac{\pi}{2} \\ \frac{\tan^{-1}(c) + \frac{1}{2}}{\pi} & -\frac{\pi}{2} < \tan^{-1}(c) < \frac{\pi}{2} \\ 1 & \tan^{-1}(c) \geq \frac{\pi}{2} \end{cases}$$

$$= \begin{cases} 0 & c < -\infty \\ \frac{\tan^{-1}(c) + \frac{1}{2}}{\pi} & -\infty < \tan^{-1}(c) < \infty \\ 1 & c > \infty \end{cases} = \frac{\tan^{-1}(c) + \frac{1}{2}}{\pi} = F_C(c)$$

Continued on next page.

Therefore we know:

$$f_C(c) = \frac{d}{dc} F_C(c) = \frac{d}{dc} \frac{\tan^{-1}(c) + \frac{1}{2}}{\pi} = \frac{1}{\pi(1+c^2)}$$

Which is the density of a Cauchy distribution.

So finally if $X \sim \text{Exponential}(1)$ then $\tan\left(-\pi\left(1 - e^{-X}\right) - \frac{\pi}{2}\right) = -\tan\left(\pi\left(1 - e^{-X}\right) + \frac{\pi}{2}\right) \sim \text{Cauchy} \quad \square$

c.

Let $Z \sim \text{Exp}(\lambda)$ then consider $W = \lceil X \rceil$ (note this means W takes values in $\{1, 2, 3, \dots\}$).

Further note that $W = k$ if and only if $X \leq k$ and $X > k - 1$.

Then for any $k \in \mathbb{Z}^+$ we have:

$$\begin{aligned} \mathbb{P}[W = k] &= \mathbb{P}[\lceil X \rceil = k] = \mathbb{P}[k - 1 < X \leq k] = \int_{k-1}^k \lambda e^{-\lambda x} dx = -e^{-\lambda x} \Big|_{k-1}^k = e^{-(k-1)\lambda} - e^{-k\lambda} \\ &= e^{-(k-1)\lambda} - e^{-(k-1+1)\lambda} = e^{-(k-1)\lambda} - e^{-(k-1)\lambda} e^{-\lambda} = e^{-(k-1)\lambda} (1 - e^{-\lambda}) \\ &= (e^{-\lambda})^{k-1} (1 - e^{-\lambda}) \end{aligned}$$

Which is precisely the probability that a $\text{Geometric}(1 - e^{-\lambda})$ random variable is equal to k for $k \in \{1, 2, 3, \dots\}$.

Therefore since $W = \lceil X \rceil$ takes the same values with the same probabilities as a $\text{Geometric}(1 - e^{-\lambda})$ random variable we know $W = \lceil X \rceil \sim \text{Geometric}(1 - e^{-\lambda}) \quad \square$

Code for Problem 1:

```
abs_diff <- function(n, p, k){
  bin_probs <- choose(n,k)*(p^k)*((1-p)^(n-k))
  poiss_probs <- ((exp(-n*p))*((n*p)^k))/(factorial(k))
  norm_probs <- (1/sqrt(2*pi*n*p*(1-p)))*exp(-((k-n*p)^2)/(2*n*p*(1-p)))
  f <- k/n
  KL <- f*log(f/p) + (1-f)*log((1-f)/(1-p))
  ent_probs <- (1/sqrt(2*pi*n*f*(1-f)))*exp(-n*KL)
  abs_diff <- data.frame(`Poisson Absolute Errors` = abs(bin_probs - poiss_probs),
    `Normal Absolute Errors` = abs(bin_probs - norm_probs),
    `Entropic Absolute Errors` = abs(bin_probs - ent_probs)
  )
  return(abs_diff)
}

rel_diff <- function(n, p, k){
  bin_probs <- choose(n,k)*(p^k)*((1-p)^(n-k))
  poiss_probs <- ((exp(-n*p))*((n*p)^k))/(factorial(k))
  norm_probs <- (1/sqrt(2*pi*n*p*(1-p)))*exp(-((k-n*p)^2)/(2*n*p*(1-p)))
  f <- k/n
  KL <- f*log(f/p) + (1-f)*log((1-f)/(1-p))
  ent_probs <- (1/sqrt(2*pi*n*f*(1-f)))*exp(-n*KL)
  rel_diff <- data.frame(`Poisson Relative Errors` = abs(bin_probs - poiss_probs)/bin_probs,
    `Normal Relative Errors` = abs(bin_probs - norm_probs)/bin_probs,
    `Entropic Relative Errors` = abs(bin_probs - ent_probs)/bin_probs
  )
  return(rel_diff)
}
```

1.a. code

```
n <- 30
p <- 0.05
k <- 0:30
abs_diff(n, p, k)
```

	Poisson.Absolute.Errors	Normal.Absolute.Errors	Entropic.Absolute.Errors
## 1	8.491396e-03	6.288532e-02	NaN
## 2	4.208071e-03	3.277279e-02	2.865137e-02
## 3	7.615308e-03	4.749378e-02	1.096979e-02
## 4	1.538910e-03	2.470382e-02	3.605901e-03
## 5	1.930467e-03	7.845186e-03	9.679277e-04
## 6	1.766931e-03	7.810490e-03	2.143120e-04
## 7	8.209923e-04	2.434696e-03	3.976033e-05
## 8	2.675847e-04	4.806307e-04	6.268669e-06
## 9	6.786047e-05	7.384760e-05	8.496632e-07
## 10	1.412178e-05	9.515639e-06	9.992828e-08
## 11	2.493920e-06	1.051824e-06	1.027277e-08
## 12	3.828577e-07	1.006534e-07	9.284106e-10
## 13	5.205110e-08	8.387780e-09	7.409012e-11
## 14	6.362462e-09	6.112552e-10	5.237982e-12
## 15	7.081188e-10	3.906518e-11	3.287950e-13
## 16	7.252526e-11	2.193133e-12	1.834906e-14
## 17	6.896636e-12	1.082138e-13	9.107894e-16
## 18	6.133846e-13	4.690382e-15	4.019292e-17

## 19	5.132796e-14	1.782894e-16	1.574848e-18
## 20	4.060356e-15	5.926516e-18	5.466524e-20
## 21	3.047997e-16	1.715570e-19	1.675528e-21
## 22	2.177936e-17	4.299675e-21	4.514869e-23
## 23	1.485157e-18	9.257674e-23	1.063399e-24
## 24	9.686240e-20	1.694769e-24	2.173290e-26
## 25	6.053980e-21	2.601619e-26	3.818433e-28
## 26	3.632400e-22	3.286255e-28	5.701307e-30
## 27	2.095617e-23	3.326169e-30	7.132860e-32
## 28	1.164232e-24	2.593504e-32	7.360840e-34
## 29	6.236956e-26	1.462502e-34	6.203044e-36
## 30	3.226012e-27	5.308539e-37	4.487914e-38
## 31	1.613006e-28	9.313226e-40	NaN

1.b. code

```
n <- 30
p <- 0.05
k <- 0:30
rel_diff(n, p, k)
```

##	Poisson.Relative.Errors	Normal.Relative.Errors	Entropic.Relative.Errors
## 1	3.956134e-02	0.29298210	NaN
## 2	1.241673e-02	0.09670249	0.084541418
## 3	2.944403e-02	0.18363122	0.042413904
## 4	1.211267e-02	0.19444231	0.028381831
## 5	4.276996e-02	0.17381198	0.021444669
## 6	1.430363e-01	0.63227346	0.017348947
## 7	3.030614e-01	0.89874474	0.014677142
## 8	5.473854e-01	0.98320354	0.012823520
## 9	9.174123e-01	0.99835288	0.011486679
## 10	1.483921e+00	0.99990583	0.010500489
## 11	2.371035e+00	0.99999689	0.009766593
## 12	3.803725e+00	0.99999994	0.009223841
## 13	6.205587e+00	1.00000000	0.008833103
## 14	1.040885e+01	1.00000000	0.008569223
## 15	1.812660e+01	1.00000000	0.008416573
## 16	3.306925e+01	1.00000000	0.008366598
## 17	6.373157e+01	1.00000000	0.008416573
## 18	1.307750e+02	1.00000000	0.008569223
## 19	2.878913e+02	1.00000000	0.008833103
## 20	6.851169e+02	1.00000000	0.009223841
## 21	1.776666e+03	1.00000000	0.009766593
## 22	5.065349e+03	1.00000000	0.010500489
## 23	1.604244e+04	1.00000000	0.011486679
## 24	5.715375e+04	1.00000000	0.012823520
## 25	2.327005e+05	1.00000000	0.014677142
## 26	1.105331e+06	1.00000000	0.017348947
## 27	6.300392e+06	1.00000000	0.021444669
## 28	4.489030e+07	1.00000000	0.028381831
## 29	4.264579e+08	1.00000000	0.042413904
## 30	6.077024e+09	1.00000000	0.084541418
## 31	1.731952e+11	1.00000000	NaN

1.c. code

```
n <- 30
p <- 0.25
k <- 0:30
abs_diff(n, p, k)
```

##	Poisson.Absolute.Errors	Normal.Absolute.Errors	Entropic.Absolute.Errors
## 1	3.745023e-04	9.548001e-04	NaN
## 2	2.362312e-03	2.148155e-03	1.509758e-04
## 3	6.924030e-03	2.799400e-03	3.660942e-04
## 4	1.203529e-02	9.512783e-04	7.621502e-04
## 5	1.249612e-02	3.802816e-03	1.295693e-03
## 6	4.646120e-03	8.218080e-03	1.816929e-03
## 7	8.737971e-03	7.738469e-03	2.134882e-03
## 8	1.975184e-02	1.723586e-03	2.131727e-03
## 9	2.198059e-02	5.202900e-03	1.829934e-03
## 10	1.536699e-02	7.910260e-03	1.363042e-03
## 11	5.034870e-03	5.645154e-03	8.874438e-04
## 12	3.450864e-03	1.547615e-03	5.079555e-04
## 13	7.510803e-03	1.259906e-03	2.567309e-04
## 14	7.686768e-03	1.983581e-03	1.149514e-04
## 15	5.874565e-03	1.495681e-03	4.569911e-05
## 16	3.721567e-03	7.971628e-04	1.615209e-05
## 17	2.046132e-03	3.299491e-04	5.077679e-06
## 18	1.003255e-03	1.104228e-04	1.419153e-06
## 19	4.471579e-04	3.054174e-05	3.521686e-07
## 20	1.838540e-04	7.073812e-06	7.742043e-08
## 21	7.055401e-05	1.382503e-06	1.502894e-08
## 22	2.550318e-05	2.287576e-07	2.564804e-09
## 23	8.744228e-06	3.202122e-08	3.825936e-10
## 24	2.858378e-06	3.772368e-09	4.952123e-11
## 25	8.940745e-07	3.702489e-10	5.510502e-12
## 26	2.683049e-07	2.978268e-11	5.210901e-13
## 27	7.740240e-08	1.915055e-12	4.128901e-14
## 28	2.150111e-08	9.472807e-14	2.698550e-15
## 29	5.759247e-09	3.385663e-15	1.440258e-16
## 30	1.489461e-09	7.782203e-17	6.599519e-18
## 31	3.723653e-10	8.625467e-19	NaN

```
rel_diff(n, p, k)
```

##	Poisson.Relative.Errors	Normal.Relative.Errors	Entropic.Relative.Errors
## 1	2.097088e+00	5.34656148	NaN
## 2	1.322816e+00	1.20289525	0.084541418
## 3	8.021846e-01	0.32432490	0.042413904
## 4	4.481840e-01	0.03542480	0.028381831
## 5	2.068200e-01	0.06293940	0.021444669
## 6	4.436348e-02	0.07847035	0.017348947
## 7	6.007286e-02	0.05320136	0.014677142
## 8	1.188183e-01	0.01036833	0.012823520
## 9	1.379744e-01	0.03265913	0.011486679
## 10	1.183829e-01	0.06093839	0.010500489
## 11	5.541030e-02	0.06212666	0.009766593
## 12	6.266342e-02	0.02810277	0.009223841
## 13	2.584172e-01	0.04334841	0.008833103
## 14	5.730215e-01	0.14786901	0.008569223

## 15	1.081940e+00	0.27546514	0.008416573
## 16	1.927728e+00	0.41292112	0.008366598
## 17	3.391593e+00	0.54691136	0.008416573
## 18	6.057917e+00	0.66676229	0.008569223
## 19	1.121563e+01	0.76604869	0.008833103
## 20	2.190430e+01	0.84277120	0.009223841
## 21	4.584970e+01	0.89842291	0.009766593
## 22	1.044118e+02	0.93654972	0.010500489
## 23	2.625296e+02	0.96137894	0.011486679
## 24	7.401769e+02	0.97685460	0.012823520
## 25	2.381354e+03	0.98615246	0.014677142
## 26	8.932828e+03	0.99157154	0.017348947
## 27	4.020123e+04	0.99464042	0.021444669
## 28	2.261365e+05	0.99629656	0.028381831
## 29	1.696030e+06	0.99703786	0.042413904
## 30	1.908035e+07	0.99691874	0.084541418
## 31	4.293080e+08	0.99444867	NaN

1.d. code

```
n <- 30
p <- 0.5
k <- 0:30
abs_diff(n, p, k)
```

##	Poisson.Absolute.Errors	Normal.Absolute.Errors	Entropic.Absolute.Errors
## 1	3.049710e-07	4.363042e-08	NaN
## 2	4.560595e-06	2.800940e-07	2.362060e-09
## 3	3.400889e-05	1.458370e-06	1.718295e-08
## 4	1.682889e-04	6.085087e-06	1.073165e-07
## 5	6.197398e-04	2.019336e-05	5.473300e-07
## 6	1.803069e-03	5.266949e-05	2.302536e-06
## 7	4.286474e-03	1.049484e-04	8.116402e-06
## 8	8.474307e-03	1.475841e-04	2.431322e-05
## 9	1.399334e-02	1.040396e-04	6.261344e-05
## 10	1.908260e-02	1.094041e-04	1.399145e-04
## 11	2.062915e-02	4.675016e-04	2.732849e-04
## 12	1.541175e-02	7.416804e-04	4.692688e-04
## 13	2.306141e-03	6.059874e-04	7.115337e-04
## 14	1.592824e-02	4.012203e-05	9.557687e-04
## 15	3.299955e-02	8.428050e-04	1.139902e-03
## 16	4.202858e-02	1.208676e-03	1.208676e-03
## 17	3.940180e-02	8.428050e-04	1.139902e-03
## 18	2.679950e-02	4.012203e-05	9.557687e-04
## 19	9.940133e-03	6.059874e-04	7.115337e-04
## 20	4.871436e-03	7.416804e-04	4.692688e-04
## 21	1.382870e-02	4.675016e-04	2.732849e-04
## 22	1.653993e-02	1.094041e-04	1.399145e-04
## 23	1.491120e-02	1.040396e-04	6.261344e-05
## 24	1.138368e-02	1.475841e-04	2.431322e-05
## 25	7.746798e-03	1.049484e-04	8.116402e-06
## 26	4.847157e-03	5.266949e-05	2.302536e-06
## 27	2.847483e-03	2.019336e-05	5.473300e-07
## 28	1.592333e-03	6.085087e-06	1.073165e-07
## 29	8.546561e-04	1.458370e-06	1.718295e-08
## 30	4.422451e-04	2.800940e-07	2.362060e-09
## 31	2.211356e-04	4.363042e-08	NaN

```
rel_diff(n, p, k)
```

##	Poisson.Relative.Errors	Normal.Relative.Errors	Entropic.Relative.Errors
## 1	3.274601e+02	4.684781e+01	NaN
## 2	1.632301e+02	1.002495e+01	0.084541418
## 3	8.394658e+01	3.599799e+00	0.042413904
## 4	4.450710e+01	1.609313e+00	0.028381831
## 5	2.428172e+01	7.911860e-01	0.021444669
## 6	1.358561e+01	3.968495e-01	0.017348947
## 7	7.751365e+00	1.897814e-01	0.014677142
## 8	4.469603e+00	7.784027e-02	0.012823520
## 9	2.567132e+00	1.908647e-02	0.011486679
## 10	1.432136e+00	8.210703e-03	0.010500489
## 11	7.372398e-01	1.670746e-02	0.009766593
## 12	3.029299e-01	1.457830e-02	0.009223841
## 13	2.862884e-02	7.522833e-03	0.008833103
## 14	1.428093e-01	3.597258e-04	0.008569223

## 15	2.436553e-01	6.222929e-03	0.008416573
## 16	2.909268e-01	8.366598e-03	0.008366598
## 17	2.909268e-01	6.222929e-03	0.008416573
## 18	2.402787e-01	3.597258e-04	0.008569223
## 19	1.233985e-01	7.522833e-03	0.008833103
## 20	9.575183e-02	1.457830e-02	0.009223841
## 21	4.942070e-01	1.670746e-02	0.009766593
## 22	1.241311e+00	8.210703e-03	0.010500489
## 23	2.735518e+00	1.908647e-02	0.011486679
## 24	6.004096e+00	7.784027e-02	0.012823520
## 25	1.400878e+01	1.897814e-01	0.014677142
## 26	3.652194e+01	3.968495e-01	0.017348947
## 27	1.115658e+02	7.911860e-01	0.021444669
## 28	4.211218e+02	1.609313e+00	0.028381831
## 29	2.109609e+03	3.599799e+00	0.042413904
## 30	1.582857e+04	1.002495e+01	0.084541418
## 31	2.374425e+05	4.684781e+01	NaN