Catam Core Projects
Computational Projects Manual
(July 2019 Edition)

21/12/2019

- Random Binary Expansions
- Programs

1 Random Binary Expansions

1.1 Question 1

For U = (U_1, U_2, ...) representing an infinite sequence of independent coin tosses, we have $U_i \sim Bernoulli(p) : P(U_i = 1) = p$, $P(U_i = 0) = 1 - p$. We also have

- $X = f(U) = \Sigma_{i=1}^{\infty} \frac{U_i}{2^i}$
- $F(x) = P(X \leq x)$
- $\hat{F}(x) = \frac{1}{N}\Sigma_{j=1}^{N} \mathbb{1}[X_j^n \leq x]$

1.1.1 Approximating F Using Monte Carlo Simulation: Programming Task

A program using this method is listed on page 11, named MCsPlot(n,N,p,step). Here we use $p = 2/3$, $n = 30$, as required. Also choose $N = 5000$ and step = 0.01.
Since $X = f(U) = \Sigma_{i=1}^{\infty} \frac{U_i}{2^i}$, it is clear that $0 \leq X \leq 1$, while equality achieves when $U_i = 0$ for all i and $U_i = 1$ for all i respectively. Hence we only need to plot the range [0, 1] for x-axis. See Fig.1 for plots of x and $\hat{F}$.

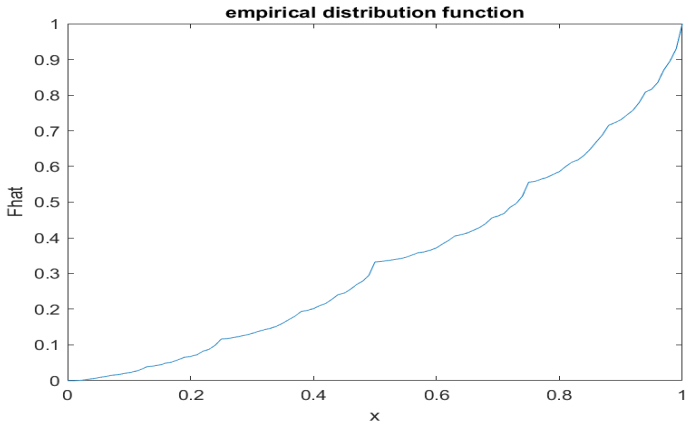

Figure 1: Empirical distribution function $\hat{F}$ (N = 5000)

We can see from Fig.1 that when N is as large as 5000, the curve is reasonably smooth. Also, choose N big enough so that the shape of the curve

does not change much when N varies a little. Otherwise, EDF (empirical distribution function) with N = 5 and N = 6 are given for contrast:

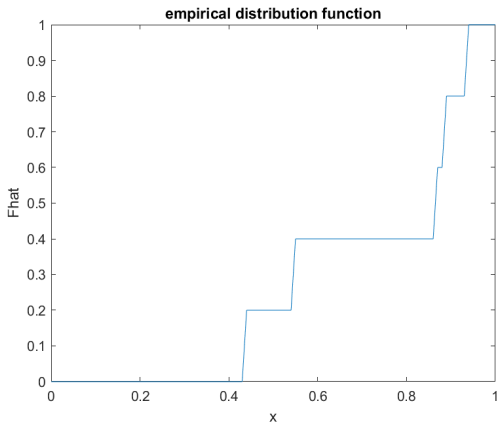

Figure 2: EDF $\hat{F}$ (N = 5)

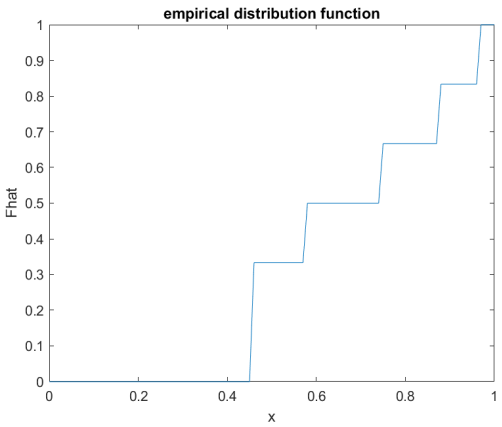

Figure 3: EDF $\hat{F}$ (N = 6)

1.2 Question 2

For x = $\Sigma_{i=1}^{n} \frac{x_i}{2^i}$, wlog can see x_i as taking value of 0 or 1, otherwise use $\lfloor 2x \rfloor$ and etc. to find the binary expansion of x with a new n'.
In order to find a formula for F(x), first find that

$$\begin{aligned} F(x+\frac{1}{2^n}) - F(x) &= P(0.x_1x_2x_3...x_n < 0.U_1U_2U_3...U_nU_{n+1}... \leq 0.x_1x_2x_3...(x_n+1)) \\ &= P(U_1 = x_1, U_2 = x_2, U_3 = x_3, ..., U_n = x_n) \\ &\times [P(U_{n+1} = 1) + P(U_{n+1} = 0, U_{n+2} = 1) + ...] \\ &= [p^{\Sigma_{i=1}^{n} \mathbb{1}[x_i=1]}(1-p)^{n-\Sigma_{i=1}^{n} \mathbb{1}[x_i=1]}] \times [p + (1-p)p + ...] \\ &= p^{\Sigma_{i=1}^{n} \mathbb{1}[x_i=1]}(1-p)^{n-\Sigma_{i=1}^{n} \mathbb{1}[x_i=1]}. \end{aligned}$$

Hence in summary, for x = $\Sigma_{i=1}^{n} \frac{x_i}{2^i} \in \{0, \frac{1}{2^n}, ..., \frac{2^n-1}{2^n}\}$, we have the recursive relation

$$\mathrm{F}(\mathrm{x}+\tfrac{1}{2^n}) = \mathrm{F}(\mathrm{x}) + p^{\Sigma_{i=1}^{n} \mathbb{1}[x_i=1]}(1-p)^{n-\Sigma_{i=1}^{n} \mathbb{1}[x_i=1]}$$

with

$$\mathrm{F}(0) = 0, \mathrm{F}(1) = 1.$$

1.3 Question 3

1.3.1 Plotting And Sampling: Programming Task

Use the formulae in §1.2, with p = 3/4 and n = 11, we get Fig.4 below. Relevant algorithm is listed on page 12, named cdfPlot(n,p).

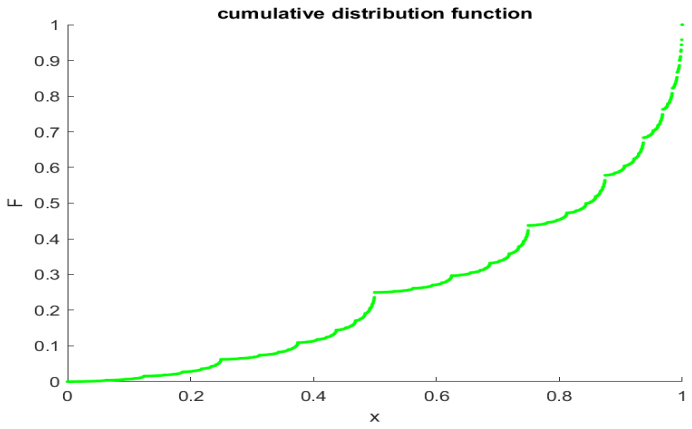

Figure 4: cdf F sampling at $\{0, \frac{1}{2^n}, ..., \frac{2^n-1}{2^n}\}$

Despite the similar shape, with a larger p, this curve is more convex than the previous one in Fig.1 and gets sharper around $\frac{i}{2^n}$, which can be seen clearly in Fig.5. Both of them are right below y = x, where p = 1/2.

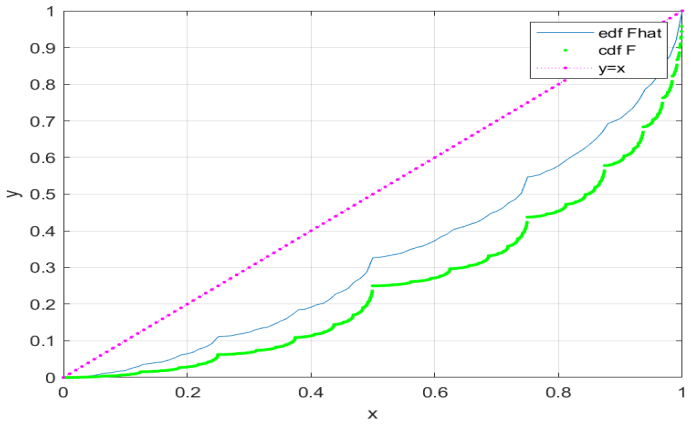

Figure 5: Compare $\hat{F}$ in Fig.1 and F in Fig.4

To calculate the complexity of the two algorithms, refer page 12 and page

13-14, for general large n and N, we get

- For MCsPlot(n,N,p,0.01), complexity = N + (n+n+n+n+1+1)×N + N×(1/step + 1) = $\mathcal{O}$(nN),
- For cdfPlot(n,p), complexity = n + 2^n + $2^n \times n = \mathcal{O}(n2^n)$.

Hence Complexity$_{MCs} \leq$ Complexity$_{cdf}$ if N $\leq 2^n$, and the other way around if otherwise.

1.4 Question 4

Proof: For any $\varepsilon > 0$, choose N s.t. $\forall$ n > N, max$\{(1-p)^n, p^n\} < \varepsilon$. Set c = $\Sigma_{i=1}^{n} \frac{c_i}{2^i} \in [0,1]$ for some n > N. Also choose $\delta = \frac{1}{2^n}$ for $|x-c| < \delta$.
For x $\geq$ c,

$$\begin{aligned} |F(x) - F(c)| &= F(x) - F(c) \\ &< F(c + \frac{1}{2^n}) - F(c) \\ &= p^{\Sigma_{i=1}^{n} \mathbb{1}[c_i=1]} (1-p)^{n - \Sigma_{i=1}^{n} \mathbb{1}[c_i=1]} \\ &\leq max\{(1-p)^n, p^n\} \\ &< \varepsilon. \end{aligned}$$

For x < $c, |F(x) - F(c)| = F(c) - F(x) < F(c) - F(c-\delta)$ and the same result follows. Hence F(x) is continuous at x = c.

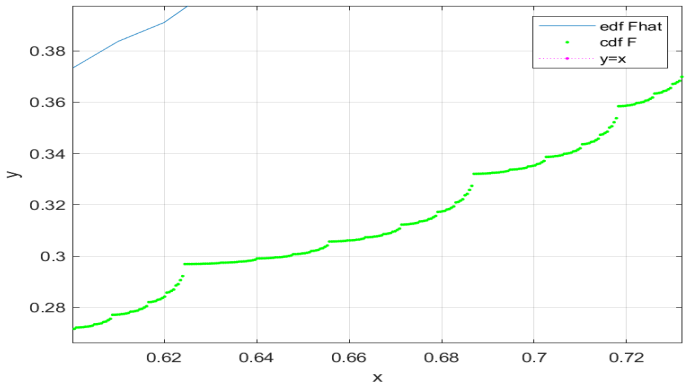

Figure 6: cdf F and edf $\hat{F}$ enlarged

To deduce whether F(x) is continuous elsewhere from the plots, try zoom in

and see in Fig.6 that dots between $\frac{i}{2^n}$ and $\frac{i+1}{2^n}$ follow a curve path. It is likely that F(x) is continuous.
For precise proof, let d be a point with infinite binary expansion. Still choose $\delta = \frac{1}{2^n}$ for $|x - d| < \delta$. For any $\varepsilon > 0$, choose N s.t. $\forall$ n > N, 2max$\{(1 - p)^n, p^n\} < \varepsilon$.
For x $\geq$ d, let c be the nearest $\frac{i}{2^n}$ below d. Then

$$\begin{aligned}
|F(x) - F(d)| &= F(x) - F(d) \\
&< F(d + \frac{1}{2^n}) - F(d) \\
&< F(c + \frac{2}{2^n}) - F(c) \\
&= p^{\Sigma_{i=1}^n \mathbb{1}[(c+\frac{1}{2^n})_i=1]}(1-p)^{n-\Sigma_{i=1}^n \mathbb{1}[c_i=1]} \\
&\leq 2max\{(1-p)^n, p^n\} \\
&< \varepsilon.
\end{aligned}$$

For x < d, let c be the nearest $\frac{i}{2^n}$ above d instead, then the same result follows. Hence F(x) is continuous everywhere within [0, 1].

1.5 Question 5

1.5.1 Analyzing differentiability of F: Programming Task

Relevant algorithm is listed on page 14-16, named limPlot(p,c,dlow,dhigh,step).

Since c = 9/16 = 1/2 + $1/2^4$, in order for $c + \delta$ to have a finite binary expansion, use $\delta = i/2^n$ ($i \in N^*$) with n $\geq$ 4 for computational convenience. Here in program we use n = 15.
To analyzing right- and left-differentiability of F(x) at c, choose δ in the range [$1/2^{15}$, $1/2^5$] and [$1/2^5$, -$1/2^{15}$] respectively with step $1/2^{15}$ in both cases. This choice of δ ensures the operating speed of the program as well as the general structure of the curve as we can see what it is like when x is very close to c. The results are in Fig.7 and Fig.8 on the next page.

We can see from Fig.7 that when $\delta \to 0^+$, $\frac{F(c+\delta)-F(c)}{\delta} \to 0^+$ despite some fluctuations, i.e., the plots suggest that the limit exists and is equal to 0, F is right-differentiable at c. From Fig.8, see that when $\delta \to 0^-$, $\frac{F(c+\delta)-F(c)}{\delta}$ increases in general to reach a local maximum before suddenly dropping to just above 0. To see whether the limit exists, also plot $\delta \in [-1/2^5, -1/2^{18}]$

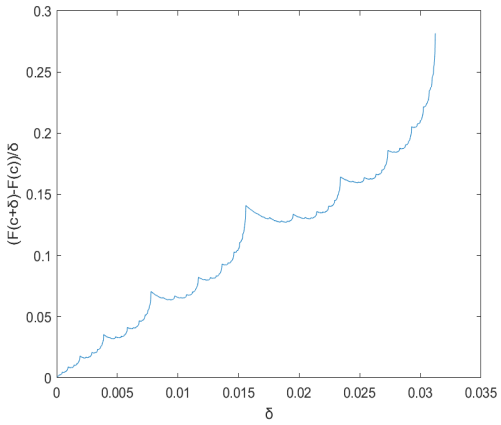

Figure 7: δ coming from right of 0

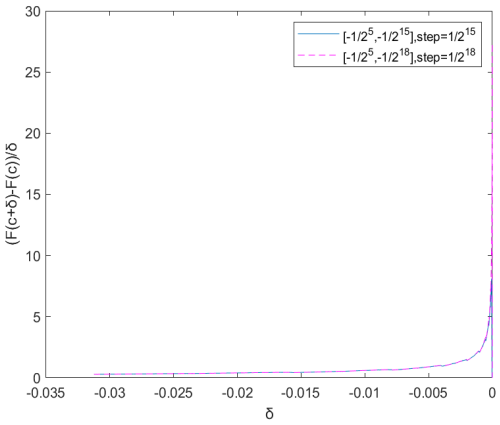

Figure 8: δ coming from left of 0

with step $1/2^{18}$ and see that when δ is closer to 0, $\frac{F(c+\delta)-F(c)}{\delta}$ gets smaller within a very small neighbourhood around zero. See also Fig.12-14 for clarity. Hence deduce that F is left-differentiable at c.

1.6 Question 6

Conjecture:

p	right-differentiable	left-differentiable
(0,1/2)	No	No
{1/2}	Yes except at 1	Yes except at 0
(1/2,1)	Yes except at 1	Yes except at 0

Proof:

- For p = 1/2, choose $\delta = 1/2^n$ so that δ goes to 0 when n becomes very large. Then the right limit is

$$\begin{aligned}\lim_{\delta\to 0^+}\frac{F(c+\delta)-F(c)}{\delta} &= \lim_{n\to\infty}\frac{F(c+1/2^n)-F(c)}{1/2^n}\\ &= \lim_{n\to\infty}\frac{1/2^n}{1/2^n}\\ &= 1.\end{aligned}$$

Choose $\delta = -1/2^n$. The left limit is

$$\begin{aligned}\lim_{\delta \to 0^-} \frac{F(c+\delta) - F(c)}{\delta} &= \lim_{n \to \infty} \frac{F(c - 1/2^n) - F(c)}{-1/2^n} \\ &= \lim_{n \to \infty} \frac{-1/2^n}{-1/2^n} \\ &= 1.\end{aligned}$$

when n becomes very large. The two limits exist and equal. F is both left- and right-differentiable at c. See Fig.9 and Fig.10 for confirmation.

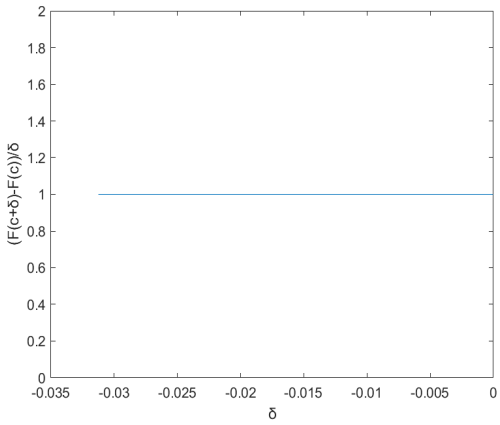

Figure 9: $\delta \to 0^-$, p = 1/2

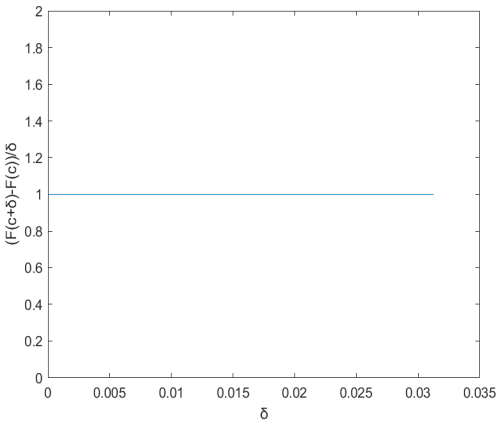

Figure 10: $\delta \to 0^+$, p = 1/2

- For p $\in$ (1/2, 1), choose $\delta = 1/2^n$. Let a = $\Sigma_{i=1}^{n} \mathbb{1}[(c + \frac{1}{2^n})_i = 1]$. Then the right limit is

$$\begin{aligned}\lim_{n \to \infty} \frac{F(c + 1/2^n) - F(c)}{1/2^n} &= \lim_{n \to \infty} \frac{p^a (1-p)^{n-a}}{1/2^n} \\ &= \lim_{n \to \infty} (2(1-p))^n (\frac{p}{1-p})^a \\ &= 0\end{aligned}$$

since $2(1-p) < 1$. Choose $\delta = -1/2^n$. The left limit is

$$\begin{aligned}\lim_{n\to\infty} \frac{F(c-1/2^n) - F(c)}{1/2^n} &= \lim_{n\to\infty} \frac{-p^a(1-p)^{n-a}}{-1/2^n} \\ &= \lim_{n\to\infty} (2(1-p))^n (\frac{p}{1-p})^a \\ &= 0\end{aligned}$$

Hence we have proved F is right- and left- differentiable at arbitrary c (except for end points) with a finite binary expansion for p $\in$ (1/2, 1). Run $limPlot(4/5, 9/16, 1/2^{15}, 1/2^5, 1/2^{15})$ and see Fig.11 that the plots confirm the right-differentiability since they aprroach to 0 as $\delta \to 0$. Also run $limPlot(4/5, 9/16, -1/2^5, -1/2^{18}, 1/2^{18})$ (smaller δ), $limPlot(4/5, 9/16, -1/2^5, -1/2^{15}, 1/2^{15})$ and see Fig.12-Fig.14 that as δ decreases the plots are closer to 0. Hence confirmed the left-differentiability.

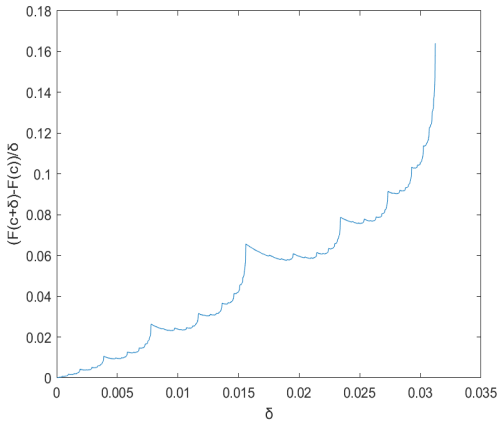

Figure 11: $\delta \to 0^+$, p = 4/5

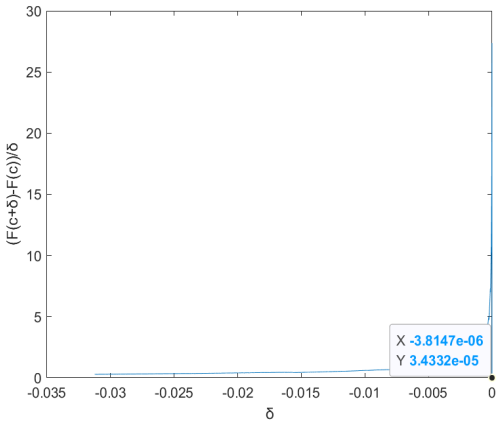

Figure 12: $\delta \to 0^-$, (smaller) $|\delta_{min}| = 1/2^{18}$, p = 4/5

- For p $\in$ (0, 1/2), choose $\delta = 1/2^n$. As before, we get

$$\begin{aligned}\frac{F(c+1/2^n) - F(c)}{1/2^n} &= \frac{p^a(1-p)^{n-a}}{1/2^n} \\ &= (2(1-p))^n (\frac{p}{1-p})^a \\ &\to \infty\end{aligned}$$

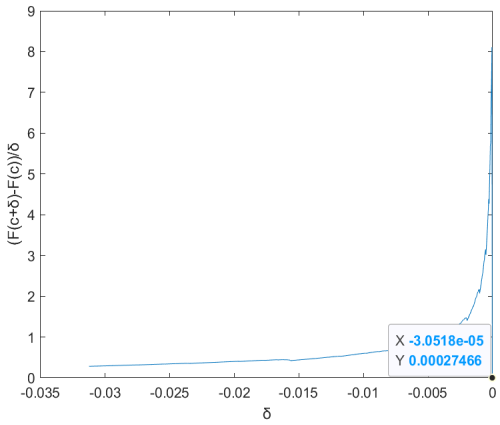

Figure 13: $\delta \to 0^-$, $|\delta_{min}| = 1/2^{15}$, p = 4/5

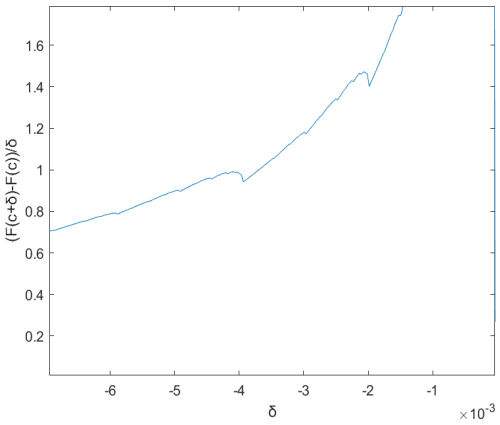

Figure 14: $\delta \to 0^-$, $|\delta_{min}| = 1/2^{15}$, p = 4/5 (enlarged Fig.13 around 0)

as n $\to \infty$. Similarly for δ coming from left. Hence F is neither right- nor left-differentiable at such c. Run $limPlot(1/4, 9/16, 1/2^{15}, 1/2^5, 1/2^{15})$ and $limPlot(1/4, 9/16, -1/2^5, -1/2^{15}, 1/2^{15})$ to confirm. See results in Fig.13 and Fig.14.

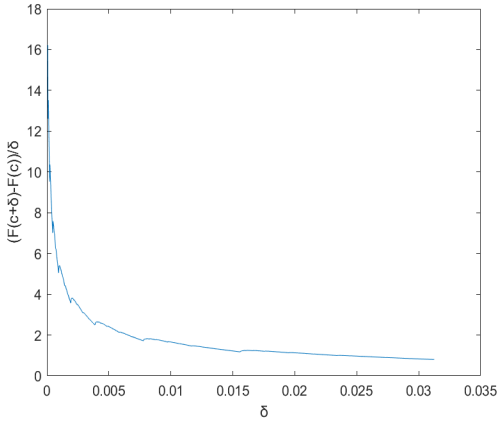

Figure 15: $\delta \to 0^-$, $|\delta_{min}| = 1/2^{15}$, p = 1/5

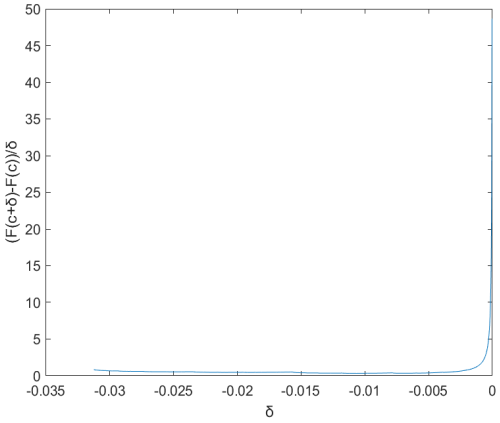

Figure 16: $\delta \to 0^-$, $|\delta_{min}| = 1/2^{15}$, p = 1/5

2 Programs

Note: Some informal programs (running just for comparison, or combining pictures, etc.) have 'return' added after excessively long texts for clarity, which is needed to be removed before tested. This situation will not happen, however, in a formal programming task that is specifically asked by the core projects.

2.1 Question 1

```
function MCsPlot(n,N,p,step)
%   Approximate F by Monte Carlo simulation.
%   Plot the empirical distribution function for p = 2/3,
%   using n = 30 and N suitably large.
%   e.g. MCsPlot(30,5000,2/3,0.01)
i=1; x=0; X=zeros(1,N); F=zeros(1,1/step+1);
while i<=N
Un=rand(1,n)>(1-p);
j=1;Xi=0;
    while j<=n
     Xi=Xi+Un(j)/(2^j);
     j=j+1;
    end
    X(1,i)=Xi;
    i=i+1;
end
X
     for j=1:(1/step+1)
          for i=1:N
               if X(1,i)<=x
                    F(1,j)=F(1,j)+1;
               else
                    F=F;
               end
          end
          x=x+step;
     end
     x=0:step:1;
```

```
    plot(x,F/N)
    xlabel('x'), ylabel('Fhat'),
    title('empirical distribution function')
end
```

2.2 Question 3

2.2.1 cdfPlot(n,p)

```
function cdfPlot(n,p)
% Use the recursive formula to plot a graph of F, for p = 3/4
% and n = 11,
% sampling F(x) at x = 0,1/2n,2/2n,3/2n,4/2n,...,1.
%   e.g. cdfPlot(11, 3/4)
xd=1/2^n; k=2; xb=zeros(1,n); F=zeros(1,2^n+1);
while xd<=1
    x=xd; i=1; j=0;
    while x~=0
      xb(1,i)=floor(2*x);
      if xb(1,i)==1
          j=j+1;
      else
          j=j;
      end
      x=2*x-xb(1,i);
      i=i+1;
    end
    F(1,k)=F(1,k-1)+p^j*(1-p)^(n-j);
    xd=xd+1/2^n;
    k=k+1;
end
X=0:1/2^n:1;
scatter(X,vpa(F,11),'g.')
xlabel('x'), ylabel('F'), title('cumulative distribution function')
end
```

2.3 Plots for Comparison of $\hat{F}$ and F

```
xd=1/2^11; k=2; xb=zeros(1,11); F1=zeros(1,2^11+1);
while xd<=1
    x1=xd; i=1; j=0;
    while x1~=0
      xb(1,i)=floor(2*x1);
      if xb(1,i)==1
          j=j+1;
      else
          j=j;
      end
      x1=2*x1-xb(1,i);
      i=i+1;
    end
    F1(1,k)=F1(1,k-1)+(3/4)^j*(1/4)^(11-j);
    xd=xd+1/2^11;
    k=k+1;
end
X1=0:1/2^11:1;
step=0.01;
i=1; x=0; X=zeros(1,5000); F=zeros(1,1/step+1);
while i<=5000
Un=rand(1,30)>(1/3);
j=1;Xi=0;
   while j<=30
    Xi=Xi+Un(j)/(2^j);
    j=j+1;
   end
   X(1,i)=Xi;
   i=i+1;
end
X
    for j=1:(1/step+1)
        for i=1:5000
            if X(1,i)<=x
                F(1,j)=F(1,j)+1;
            else
```

```
                F=F;
            end
        end
        x=x+step;
    end
    x=0:step:1;
    x3=x
    Y=x3
    plot(x,F/5000,X1,vpa(F1,11),'g.',Y,x3, 'm:.')
    xlabel('x'), ylabel('y')
    legend('edf Fhat' , 'cdf F', 'y=x')
    grid on
```

2.4 Question 5

Programs for limPlot and plotting.

2.5 limPlot(p,c,dlow,dhigh,step)

```
function limPlot(p,c,dlow,dhigh,step)
%Plot (F(c + Î´) - F(c))/Î´ against Î´ for a suitable range of Î´
% for which c+Î´ has a fnite binary expansion.
%   e.g. limPlot(3/4,9/16,1/2^15,1/2^5,1/2^15)
%        limPlot(3/4,9/16,-1/2^5,-1/2^15,1/2^15)
n=(dhigh-dlow)/step+1;
xd=step; k=2; xb=zeros(1,n+1); F=xb; G=zeros(1,n); X=G; l=1;
m=log2(1/step);
while xd<=c+dhigh+step
    x=xd; i=1; j=0;
    while x~=0
      xb(1,i)=floor(2*x);
      if xb(1,i)==1
          j=j+1;
      else
          j=j;
      end
      x=2*x-xb(1,i);
```

```
        i=i+1;
    end
    F(1,k)=F(1,k-1)+p^j*(1-p)^(m-j);
    if xd==c
        Fc=F(1,k);
    end
    if dlow>0
        if xd<c+step
            G=G;
        elseif xd<=c+dhigh
            G(1,l)=(F(1,k)-Fc)/(xd-c);
            X(1,l)=xd-c;
            l=l+1;
        else
            G=G;
        end
        xd=xd+step;
        k=k+1;
    else
        if xd<c+dlow
            G=G;
        elseif xd<=c+dhigh
            G(1,l)=F(1,k);
            X(1,l)=xd-c;
            l=l+1;
        else
            G=G;
        end
        xd=xd+step;
        k=k+1;
    end
end
if dlow>0
    plot(vpa(X) ,vpa(G));
else
    for l=1:n
        G(1,l)=(G(1,l)-Fc)/X(1,l);
    end
```

```
    plot(vpa(X)  ,vpa(G));
end
xlabel('Î´'), ylabel('(F(c+Î´)-F(c))/Î´')
end
```

2.6 Plotting

```
p=3/4;c=9/16;dlow=-1/2^5;dhigh=-1/2^15;step=1/2^15;
n=(dhigh-dlow)/step+1;
xd=step; k=2; xb=zeros(1,n+1); F=xb; G=zeros(1,n); X=G; l=1;
m=log2(1/step);
while xd<=c+dhigh+step
    x=xd; i=1; j=0;
    while x~=0
      xb(1,i)=floor(2*x);
      if xb(1,i)==1
          j=j+1;
      else
          j=j;
      end
      x=2*x-xb(1,i);
      i=i+1;
    end
    F(1,k)=F(1,k-1)+p^j*(1-p)^(m-j);
    if xd==c
        Fc=F(1,k);
    end
        if xd<c+dlow
            G=G;
        elseif xd<=c+dhigh
            G(1,l)=F(1,k);
            X(1,l)=xd-c;
            l=l+1;
        else
            G=G;
        end
        xd=xd+step;
```

```
        k=k+1;
end

    for l=1:n
        G(1,l)=(G(1,l)-Fc)/X(1,l);
    end

p=3/4;c=9/16;dlow=-1/2^5;dhigh=-1/2^18;step=1/2^18;
n=(dhigh-dlow)/step+1;
xd=step; k=2; xb=zeros(1,n+1); F1=xb; G1=zeros(1,n); X1=G1; l=1;
m=log2(1/step);
while xd<=c+dhigh+step
    x=xd; i=1; j=0;
    while x~=0
      xb(1,i)=floor(2*x);
      if xb(1,i)==1
          j=j+1;
      else
          j=j;
      end
      x=2*x-xb(1,i);
      i=i+1;
    end
    F1(1,k)=F1(1,k-1)+p^j*(1-p)^(m-j);
    if xd==c
        Fc=F1(1,k);
    end
        if xd<c+dlow
            G1=G1;
        elseif xd<=c+dhigh
            G1(1,l)=F1(1,k);
            X1(1,l)=xd-c;
            l=l+1;
        else
            G1=G1;
        end
        xd=xd+step;
        k=k+1;
```

```
end

    for l=1:n
        G1(1,l)=(G1(1,l)-Fc)/X1(1,l);
    end
    plot(vpa(X) ,vpa(G),vpa(X1),vpa(G1),'m--');
xlabel('Î´'), ylabel('(F(c+Î´)-F(c))/Î´')
legend('[-1/2^5,-1/2^{15}],step=1/2^{15}'
,'[-1/2^5,-1/2^{18}],step=1/2^{18}')
```