8. Estimating the CDF and Statistical Functionals

8.1 Empirical distribution function

The **empirical distribution function** $\hat{F}_n$ is the CDF that puts mass $1/n$ at each data point X_i. Formally,

$$\begin{aligned}\hat{F}_n(x) &= \frac{\sum_{i=1}^{n} I\left(X_i \leq x\right)}{n} \qquad (1)\\ &= \frac{\#|\text{observations less than or equal to x}|}{n} \qquad (2)\end{aligned}$$

where

$$I\left(X_i \leq x\right) = \begin{cases} 1 & \text{if } X_i \leq x \\ 0 & \text{if } X_i > x \end{cases} \qquad (3)$$

Theorem 8.3. At any fixed value of x,

$$\mathbb{E}\left(\hat{F}_n(x)\right) = F(x) \quad \text{and} \quad \mathbb{V}\left(\hat{F}_n(x)\right) = \frac{F(x)(1-F(x))}{n} \qquad (4)$$

Thus,

$$\text{MSE} = \frac{F(x)(1-F(x))}{n} \rightarrow 0$$

and hence, $\hat{F}_n(x) \xrightarrow{\text{P}} F(x)$.

Theorem 8.4 (Glivenko-Cantelli Theorem). Let $X_1, \ldots, X_n \sim F$. Then

$$\sup_x |\hat{F}_n(x) - F(x)| \xrightarrow{\text{P}} 0$$

(actually, $\sup_x |\hat{F}_n(x) - F(x)|$ converges to 0 almost surely.)

8.2 Statistical Functionals

A **statistical functional** $T(F)$ is any function of F. Examples are the mean $\mu = \int x dF(x)$, the variance $\sigma^2 = \int (x-\mu)^2 dF(x)$ and the median

$m = F^{-1}(1/2)$.

The **plug-in estimator** of $\theta = T(F)$ is defined by

$$\hat{\theta_n} = T(\hat{F_n})$$

In other words, just plug in $\hat{F_n}$ for the unknown F.

A functional of the form $\int r(x) dF(x)$ is called a **linear functional**. Recall that $\int r(x) dF(x)$ is defined to be $\int r(x) f(x) d(x)$ in the continuous case and $\sum_j r(x_j) f(x_j)$ in the discrete.

The plug-in estimator for the linear functional $T(F) = \int r(x) dF(x)$ is:

$$T(\hat{F_n}) = \int r(x) d\hat{F_n}(x) = \frac{1}{n} \sum_{i=1}^n r(X_i)$$

We have:

$$T(\hat{F_n}) \approx N\left(T(F), \hat{\text{se}}\right)$$

An approximate $1 - \alpha$ confidence interval for $T(F)$ is then

$$T(\hat{F_n}) \pm z_{\alpha/2} \hat{\text{se}}$$

We call this the **Normal-based interval**.

8.3 Technical Appendix

Theorem 8.12 (Dvoretsky-Kiefer-Wolfowitz (DKW) inequality). Let $X_1, \dots, X_n$ be iid from F. Then, for any $\epsilon > 0$,

$$\mathbb{P} \left(\sup_x |F(x) - \hat{F_n}(x)| > \epsilon \right) \leq 2e^{-2n\epsilon^2}$$

From the DKW inequality, we can construct a confidence set. Let $\epsilon_n^2 = \log(2/\alpha)/(2n)$, $L(x) = \max\{ \hat{F_n}(x) - \epsilon_n, \; 0\}$ and $U(x) = \min\{ \hat{F_n}(x) + \epsilon_n, 1 \}$. It follows that for any F,

$$\mathbb{P}(F \in C_n) \geq 1 - \alpha$$

To summarize:

A $1-\alpha$ nonparametric confidence band for F is $(L(x),\ U(x))$ where

$$L(x) = \max\{\hat{F}_n(x) - \epsilon_n,\ 0\} \tag{5}$$
$$U(x) = \min\{\hat{F}_n(x) + \epsilon_n,\ 1\} \tag{6}$$
$$\epsilon_n = \sqrt{\frac{1}{2n}\log\left(\frac{2}{\alpha}\right)} \tag{7}$$

8.5 Exercises

Exercise 8.5.1. Prove Theorem 8.3.

Solution. We have:

$$\hat{F}_n(x) = \frac{\sum_{i=1}^n I\left(X_i \leq x\right)}{n}$$

where

$$I\left(X_i \leq x\right) = \begin{cases} 1 & \text{if } X_i \leq x \\ 0 & \text{if } X_i > x \end{cases} \tag{8}$$

Thus,

$$\mathbb{E}(\hat{F}_n(x)) = n^{-1}\sum_{i=1}^n \mathbb{E}(I\left(X_i \leq x\right)) \tag{9}$$
$$= n^{-1}\sum_{i=1}^n \mathbb{P}\left(X_i \leq x\right) \tag{10}$$
$$= n^{-1}\sum_{i=1}^n F(x) \tag{11}$$
$$= F(x) \tag{12}$$

$$
\begin{aligned}
\mathbb{E}(\hat{F}_n(x)^2) &= n^{-2}\mathbb{E}\left(\sum_{i=1}^n I\left(X_i \leq x\right)\right)^2 && (13)\\
&= n^{-2}\mathbb{E}\left(\sum_{i=1}^n I(X_i \leq x)^2 + \sum_{i=1}^n \sum_{j=1, j\neq i}^n I\left(X_i \leq x\right) I\left(X_j \leq x\right)\right) && (14)\\
&= n^{-2}\left(\sum_{i=1}^n \mathbb{E}\left(I(X_i \leq x)^2\right) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n \mathbb{E}\left(I\left(X_i \leq x\right) I\left(X_j \leq x\right)\right)\right) && (15)\\
&= n^{-2}\left(\sum_{i=1}^n \mathbb{E}\left(I\left(X_i \leq x\right)\right) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n \mathbb{E}\left(I\left(X_i \leq x\right)\right) \mathbb{E}\left(I\left(X_j \leq x\right)\right)\right) && (16)\\
&= n^{-2}\left(\sum_{i=1}^n \mathbb{P}\left(X_i \leq x\right) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n \mathbb{P}\left(X_i \leq x\right) \mathbb{P}\left(X_j \leq x\right)\right) && (17)\\
&= n^{-2}\left(\sum_{i=1}^n F(x) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n F(x)^2\right) && (18)\\
&= n^{-2}\left(nF(x) + (n^2 - n)F(x)^2\right) && (19)\\
&= n^{-1}(F(x) + (n-1)F(x)^2) && (20)
\end{aligned}
$$

Therefore,

$$\mathbb{V}(\hat{F}_n(x)) = \mathbb{E}(\hat{F}_n(x)^2) - \mathbb{E}(\hat{F}_n(x))^2 = F(x)/n + (1 - 1/n)F(x)^2 - F(x)^2 = \frac{F(x)(1-F(x))}{n}$$

Finally,

$$\text{MSE} = (\text{bias}(F(x)))^2 + \mathbb{V}(F(x)) = (\mathbb{E}(\hat{F}_n(x)) - F(x))^2 + \mathbb{V}(F(x)) = \mathbb{V}(F(x)) = \frac{F(x)(1-F(x))}{n} \to 0$$

Exercise 8.5.2. Let $X_1, \dots, X_n \sim \text{Bernoulli}(p)$ and let $Y_1, \dots, Y_m \sim \text{Bernoulli}(q)$.

- Find the plug-in estimator and estimated standard error for p.
- Find an approximate 90-percent confidence interval for p.
- Find the plug-in estimator and estimated standard error for $p - q$.
- Find an approximate 90-percent confidence interval for $p - q$.

Solution.

(a)

p is the mean of $\text{Bernoulli}(p)$, so its plugin estimator is $\hat{p} = \mathbb{E}(\hat{F}_n) = n^{-1}\sum_{i=1}^n X_i = \overline{X}_n$.

$\sqrt{p(1-p)}$ is the standard error of $\text{Bernoulli}(p)$, so its plugin estimator is $\sqrt{\hat{p}(1-\hat{p})} = \sqrt{\overline{X}_n(1-\overline{X}_n)}$.

(b)

The 90-percent confidence interval for p is $\hat{p} \pm z_{5\%}\hat{se}(\hat{p}) = \overline{X}_n \pm z_{5\%}\sqrt{\overline{X}_n(1-\overline{X}_n)}$.

(c)

The plug-in estimator for $\theta = p - q$ is $\hat{\theta} = \hat{p} - \hat{q} = \overline{X}_n - \overline{Y}_m$.

The standard error of $\hat{\theta}$ is

$$\text{se} = \sqrt{\mathbb{V}(\hat{p} - \hat{q})} = \sqrt{\mathbb{V}(\hat{p}) + \mathbb{V}(\hat{q})} = \sqrt{\hat{p}(1-\hat{p}) + \hat{q}(1-\hat{q})} = \sqrt{\overline{X}_n(1-\overline{X}_n) + \overline{Y}_m(1-\overline{Y}_m)}$$

(d)

The 90-percent confidence interval for $\theta = p - q$ is $\hat{\theta} \pm z_{5\%}\hat{se}(\hat{\theta}) = \overline{X}_n - \overline{Y}_m \pm z_{5\%}\sqrt{\overline{X}_n(1-\overline{X}_n) + \overline{Y}_m(1-\overline{Y}_m)}$

Exercise 8.5.3. (Computer Experiment) Generate 100 observations from a $N(0, 1)$ distribution. Compute a 95 percent confidence band for the CDF F. Repeat this 1000 times and see how often the confidence band contains the true function. Repeat using data from a Cauchy distribution.

In [1]:

```
import math
import numpy as np
import pandas as pd
from scipy.stats import norm, cauchy
import matplotlib.pyplot as plt

from tqdm import tqdm_notebook
```

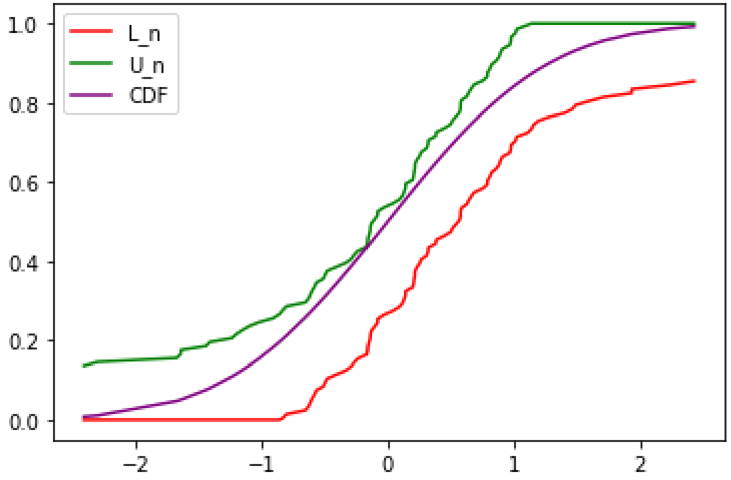
In [2]:

```
# One iteration wtih Normal distribution
```

```
n = 100
alpha = 0.05
r = norm.rvs(size=n)
epsilon = math.sqrt((1 / (2 * n)) * math.log(2 / alpha))

F_n = lambda x : sum(r < x) / n
L_n = lambda x : max(F_n(x) - epsilon, 0)
U_n = lambda x : min(F_n(x) + epsilon, 1)

xx = sorted(r)

df = pd.DataFrame({
    'x': xx,
    'F_n': np.array(list(map(F_n, xx))),
    'U_n': np.array(list(map(U_n, xx))),
    'L_n': np.array(list(map(L_n, xx))),
    'CDF': np.array(list(map(norm.cdf, xx)))
})
df['in_bounds'] = (df['U_n'] >= df['CDF']) & (df['CDF'] >= df['L_n'])

plt.plot( 'x', 'L_n', data=df, color='red')
plt.plot( 'x', 'U_n', data=df, color='green')
plt.plot( 'x', 'CDF', data=df, color='purple')
plt.legend()
```

Out[2]:

```
<matplotlib.legend.Legend at 0x26bbd31ea90>
```



In [3]:

```
# 1000 iterations with Normal distribution
```

```
bounds = []
for k in tqdm_notebook(range(1000)):
    n = 100
    alpha = 0.05
    r = norm.rvs(size=n)
    epsilon = math.sqrt((1 / (2 * n)) * math.log(2 / alpha))

    F_n = lambda x : sum(r < x) / n
    L_n = lambda x : max(F_n(x) - epsilon, 0)
    U_n = lambda x : min(F_n(x) + epsilon, 1)

    # xx = sorted(r)
    xx = r # No need to sort without plotting

    df = pd.DataFrame({
        'x': xx,
        'F_n': np.array(list(map(F_n, xx))),
        'U_n': np.array(list(map(U_n, xx))),
        'L_n': np.array(list(map(L_n, xx))),
        'CDF': np.array(list(map(norm.cdf, xx)))
    })
    all_in_bounds = ((df['U_n'] >= df['CDF']) & (df['CDF'] >= df['L_n'])).all()
    bounds.append(all_in_bounds)

print('Average fraction in bounds: %.3f' % np.array(bounds).mean())
```

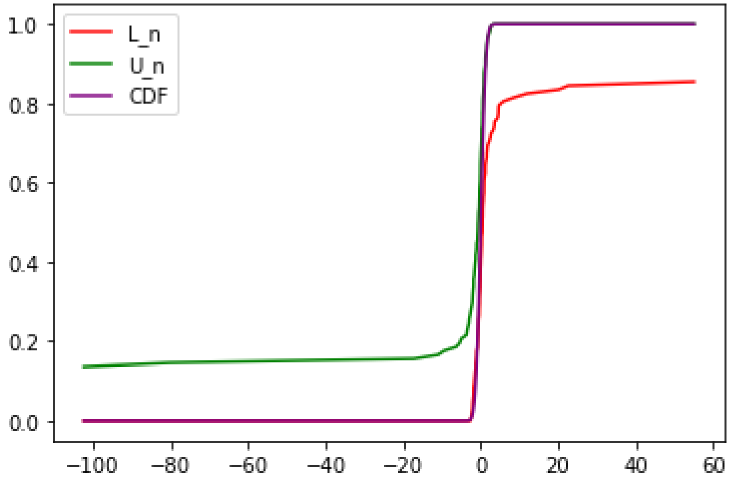
```
Average fraction in bounds: 0.963
```

In [4]:

```
# One iteration wtih Cauchy distribution

n = 100
alpha = 0.05
r = cauchy.rvs(size=n)
epsilon = math.sqrt((1 / (2 * n)) * math.log(2 / alpha))

F_n = lambda x : sum(r < x) / n
L_n = lambda x : max(F_n(x) - epsilon, 0)
U_n = lambda x : min(F_n(x) + epsilon, 1)

xx = sorted(r)

df = pd.DataFrame({
    'x': xx,
    'F_n': np.array(list(map(F_n, xx))),
    'U_n': np.array(list(map(U_n, xx))),
    'L_n': np.array(list(map(L_n, xx))),
    'CDF': np.array(list(map(norm.cdf, xx)))
})
df['in_bounds'] = (df['U_n'] >= df['CDF']) & (df['CDF'] >= df['L_n'])

plt.plot( 'x', 'L_n', data=df, color='red')
plt.plot( 'x', 'U_n', data=df, color='green')
plt.plot( 'x', 'CDF', data=df, color='purple')
plt.legend()
```

Out[4]:

```
<matplotlib.legend.Legend at 0x26bbe413eb8>
```



In [5]:

```
# 1000 iterations with Cauchy distribution

bounds = []
for k in tqdm_notebook(range(1000)):
    n = 100
    alpha = 0.05
    r = cauchy.rvs(size=n)
    epsilon = math.sqrt((1 / (2 * n)) * math.log(2 / alpha))

    F_n = lambda x : sum(r < x) / n
    L_n = lambda x : max(F_n(x) - epsilon, 0)
    U_n = lambda x : min(F_n(x) + epsilon, 1)

    # xx = sorted(r)
    xx = r # No need to sort without plotting

    df = pd.DataFrame({
        'x': xx,
        'F_n': np.array(list(map(F_n, xx))),
        'U_n': np.array(list(map(U_n, xx))),
        'L_n': np.array(list(map(L_n, xx))),
        'CDF': np.array(list(map(norm.cdf, xx)))
    })
    all_in_bounds = ((df['U_n'] >= df['CDF']) & (df['CDF'] >= df['L_n'])).all()
    bounds.append(all_in_bounds)

print('Average fraction in bounds: %.3f' % np.array(bounds).mean())
```

```
Average fraction in bounds: 0.204
```

Exercise 8.5.4. Let $X_1, \dots, X_n \sim F$ and let $\hat{F}_n(x)$ be the empirical distribution function. For a fixed x, use the central limit theorem to find the limiting distribution of $\hat{F}_n(x)$.

Solution.

We have:

$$\hat{F}_n(x) = \frac{\sum_{i=1}^n I\left(X_i \leq x\right)}{n}$$

where

$$I\left(X_i \leq x\right) = \begin{cases} 1 & \text{if } X_i \leq x \\ 0 & \text{if } X_i > x \end{cases} \tag{21}$$

Let $Y_i = I\left(X_i \leq x\right)$ for some fixed x. From the central limit theorem,

$$\sqrt{n}(\overline{Y}_n - \mu_Y) \leadsto N(0, \sigma_Y^2) \tag{22}$$

$$\overline{Y}_n \leadsto N(\mu_Y, \sigma_Y^2/n) \tag{23}$$

We can estimate the mean μ_Y as $\mathbb{E}(\hat{\mu_Y}) = \mathbb{E}(\overline{Y}_n) = n^{-1}\sum_{i=1}^n \mathbb{E}(I(X_i \leq x)) = n^{-1}\sum_{i=1}^n F(x) = \hat{F}_n(x)$.

We can estimate the variance σ_Y^2 as

$\mathbb{E}(\hat{\sigma_Y}^2) = \mathbb{E}(\mathbb{V}(\overline{Y}_n)) = n^{-1} \sum_{i=1}^n \mathbb{E}((Y_i - \overline{Y}_n)^2) = n^{-1} \sum_{i=1}^n \left(\mathbb{E}(Y_i^2) - 2\mathbb{E}(Y_i \overline{Y}_n) + \mathbb{E}(\overline{Y}_n^2) \right) \leq n^{-1} \sum_{i=1}^n \left(\mathbb{E}(Y_i) + \mathbb{E}(\overline{Y}_n^2) \right) \leq 2$

Therefore, for large n, the limiting distribution has variance that goes to 0 -- so $\overline{Y}_n \rightsquigarrow \mu_Y$, or $I\left(X_i \leq x\right) \rightsquigarrow F(x)$ for every x. Then,

$\hat{F}_n(x) \rightsquigarrow n^{-1} \sum_{i=1}^n F(x) = F(x)$,

and, as expected, F is the limiting distribution of F_n.

Exercise 8.5.5. Let x and y be two distinct points. Find $\text{Cov}(\hat{F}_n(x), \hat{F}_n(y))$.

Solution.

We have:

$$\begin{aligned} \text{Cov}(\hat{F}_n(x), \hat{F}_n(y)) &= \mathbb{E}(\hat{F}_n(x)\hat{F}_n(y)) - \mathbb{E}(\hat{F}_n(x))\mathbb{E}(\hat{F}_n(y)) && (24) \\ &= \mathbb{E}(\hat{F}_n(x)\hat{F}_n(y)) - F(x)F(y) && (25) \end{aligned}$$

But:

$$\hat{F}_n(x)\hat{F}_n(y) = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n I(X_i \leq x) I(X_j \leq y)$$

so

$$\begin{aligned}
\mathbb{E}(\hat{F}_n(x)\hat{F}_n(y)) &= \frac{1}{n^2}\sum_{i=1}^n\sum_{j=1}^n \mathbb{E}(I(X_i \leq x)I(X_j \leq y)) && (26)\\
&= \frac{1}{n^2}\sum_{i=1}^n\sum_{j=1}^n \mathbb{P}(X_i \leq x, X_j \leq y) && (27)\\
&= \frac{1}{n^2}\sum_{i=1}^n\sum_{j=1}^n \mathbb{P}(X_i \leq x | X_j \leq y)\mathbb{P}(X_j \leq y) && (28)\\
&= \frac{1}{n^2}\left(\sum_{i=1}^n F(\min\{x, y\}) + \sum_{i=1}^n\sum_{j=1, j\neq i}^n F(x)F(y)\right) && (29)\\
&= \frac{1}{n}F(\min\{x, y\}) + \left(1 - \frac{1}{n}\right)F(x)F(y) && (30)
\end{aligned}$$

Therefore, assuming $x \leq y$,

$$\begin{aligned}
\text{Cov}(\hat{F}_n(x), \hat{F}_n(y)) &= \mathbb{E}(\hat{F}_n(x)\hat{F}_n(y)) - \mathbb{E}(\hat{F}_n(x))\mathbb{E}(\hat{F}_n(y)) && (31)\\
&= \mathbb{E}(\hat{F}_n(x)\hat{F}_n(y)) - F(x)F(y) && (32)\\
&= \frac{1}{n}F(\min\{x, y\}) + \left(1 - \frac{1}{n}\right)F(x)F(y) - F(x)F(y) && (33)\\
&= \frac{F(x)(1 - F(y))}{n} && (34)
\end{aligned}$$

Exercise 8.5.6. Let $X_1, \dots, X_n \sim F$ and let $\hat{F}$ be the empirical distribution function. Let $a < b$ be fixed numbers and define $\theta = T(F) = F(b) - F(a)$. Let $\hat{\theta} = T(\hat{F}_n) = \hat{F}_n(b) - \hat{F}_n(a)$.

- Find the estimated standard error of $\hat{\theta}$.
- Find an expression for an approximate $1 - \alpha$ confidence interval for θ.

Solution.

(a)

The estimated mean for $\hat{\theta}$ is $\mathbb{E}(\hat{\theta}) = \mathbb{E}(\hat{F}_n(b) - \hat{F}_n(a)) = \mathbb{E}(\hat{F}_n(b)) - \mathbb{E}(\hat{F}_n(a)) = F(b) - F(a) = \theta$.

The estimated variance for $\hat{\theta}$ is

$$\mathbb{V}(\hat{\theta}) = \mathbb{E}(\hat{\theta}^2) - \mathbb{E}(\hat{\theta})^2$$

But

$$\begin{aligned}
\mathbb{E}(\hat{\theta}^2) &= \mathbb{E}((\hat{F}_n(b) - \hat{F}_n(a))^2) && (35)\\
&= \mathbb{E}(\hat{F}_n(a)^2 + \hat{F}_n(b)^2 - 2\hat{F}_n(a)\hat{F}_n(b)) && (36)\\
&= \mathbb{E}(\hat{F}_n(a)^2) + \mathbb{E}(\hat{F}_n(b)^2) - 2\mathbb{E}(\hat{F}_n(a)\hat{F}_n(b)) && (37)
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}(\hat{F}_n(a)^2) &= \mathbb{E}\left(\left(\frac{1}{n}\sum_{i=1}^n I(X_i \le a)\right)^2\right) && (38)\\
&= \frac{1}{n^2}\left(\sum_{i=1}^n \mathbb{E}\left(I(X_i \le a)^2\right) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n \mathbb{E}\left(I(X_i \le a) I(X_j \le a)\right)\right) && (39)\\
&= \frac{1}{n^2}\left(\sum_{i=1}^n \mathbb{E}\left(I(X_i \le a)\right) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n \mathbb{E}\left(I(X_i \le a) I(X_j \le a)\right)\right) && (40)\\
&= \frac{1}{n^2}\left(\sum_{i=1}^n \mathbb{P}\left(X_i \le a\right) + \sum_{i=1}^n \sum_{j=1, j\neq i}^n \mathbb{P}\left(X_i \le a, X_j \le a\right)\right) && (41)\\
&= \frac{1}{n^2}\left(nF(a) + n(n-1)F(a)^2\right) && (42)\\
&= F(a)\frac{1}{n} + F(a)^2\left(1 - \frac{1}{n}\right) && (43)\\
\mathbb{E}(\hat{F}_n(b)^2) &= F(b)\frac{1}{n} + F(b)^2\left(1 - \frac{1}{n}\right) && (44)
\end{aligned}$$

From the previous exercise,

$$\mathbb{E}(\hat{F}_n(a)\hat{F}_n(b)) = \frac{1}{n}F(a) + \left(1 - \frac{1}{n}\right) F(a)F(b)$$

Putting it together,

$$
\begin{aligned}
\mathbb{E}(\hat{\theta}^2) &= \mathbb{E}(\hat{F}_n(a)^2) + \mathbb{E}(\hat{F}_n(b)^2) - 2\mathbb{E}(\hat{F}_n(a)\hat{F}_n(b)) && (45) \\
&= F(a)\frac{1}{n} + F(a)^2\left(1 - \frac{1}{n}\right) + F(b)\frac{1}{n} + F(b)^2\left(1 - \frac{1}{n}\right) - 2\left(\frac{1}{n}F(a) + \left(1 - \frac{1}{n}\right)F(a)F(b)\right) && (46) \\
&= \frac{1}{n}(F(b) - F(a)) + \left(1 - \frac{1}{n}\right)(F(b) - F(a))^2 && (47) \\
&= \frac{1}{n}\theta + \left(1 - \frac{1}{n}\right)\theta^2 && (48) \\
\mathbb{V}(\hat{\theta}) &= \mathbb{E}(\hat{\theta}^2) - \mathbb{E}(\hat{\theta})^2 && (49) \\
&= \frac{1}{n}\theta + \left(1 - \frac{1}{n}\right)\theta^2 - \theta^2 && (50) \\
&= \frac{\theta(1-\theta)}{n} && (51)
\end{aligned}
$$

Finally, the estimated standard error is

$$\text{se}(\hat{\theta}) = \sqrt{\mathbb{V}(\hat{\theta})} = \sqrt{\frac{\hat{\theta}(1-\hat{\theta})}{n}}$$

(b)

An approximate $1 - \alpha$ confidence interval is

$$\hat{\theta} \pm z_{\alpha/2}\text{se}(\hat{\theta}) = \hat{\theta} \pm z_{\alpha/2}\sqrt{\frac{\hat{\theta}(1-\hat{\theta})}{n}}$$

Exercise 8.5.7. Data on the magnitudes of earthquakes near Fiji are available on the course website.

- Estimate the CDF.
- Compute and plot a 95% confidence envelope for F.
- Find an approximate 95% confidence interval for F(4.9) - F(4.3).

In [6]:

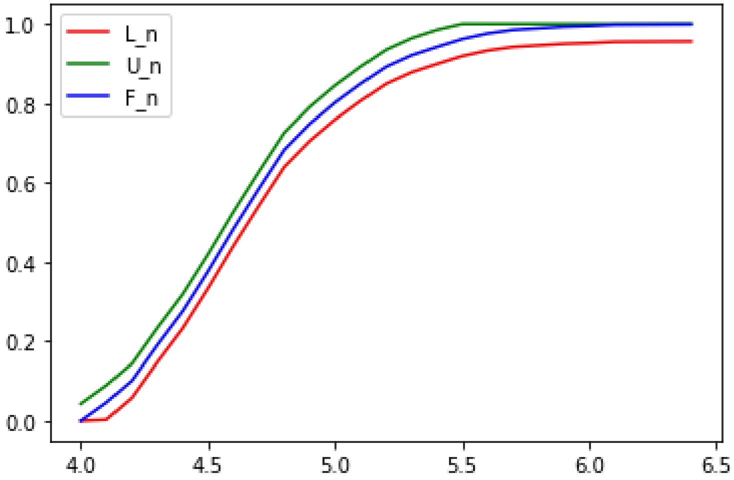
```
import pandas as pd

data = pd.read_csv('data/fijiquakes.csv', sep='\t')
r = np.array(data['mag'])
```

In [7]:

```
n = len(r)
alpha = 0.05
epsilon = math.sqrt((1 / (2 * n)) * math.log(2 / alpha))

F_n = lambda x : sum(r < x) / n
L_n = lambda x : max(F_n(x) - epsilon, 0)
U_n = lambda x : min(F_n(x) + epsilon, 1)

xx = sorted(r)

df = pd.DataFrame({
    'x': xx,
    'F_n': np.array(list(map(F_n, xx))),
    'U_n': np.array(list(map(U_n, xx))),
    'L_n': np.array(list(map(L_n, xx)))
})

plt.plot( 'x', 'L_n', data=df, color='red')
plt.plot( 'x', 'U_n', data=df, color='green')
plt.plot( 'x', 'F_n', data=df, color='blue')
plt.legend()
```

Out[7]:

```
<matplotlib.legend.Legend at 0x26bbe4be828>
```



In [8]:

```
# Now to find the confidence interval, using the result from 8.5.6:

import math
from scipy.stats import norm
```

```
z_95 = norm.ppf(.975)
theta = F_n(4.9) - F_n(4.3)
se = math.sqrt(theta * (1 - theta) / n)

print('95%% confidence interval: (%.3f, %.3f)' % ((theta - z_95 * se), (theta + z_95 * se)))
```

```
95% confidence interval: (0.526, 0.588)
```

Exercise 8.5.8. Get the data on eruption times and waiting times between eruptions of the old faithful geyser from the course website.

- Estimate the mean waiting time and give a standard error for the estimate.
- Also, give a 90% confidence interval for the mean waiting time.
- Now estimate the median waiting time.

In the next chapter we will see how to get the standard error for the median.

In [9]:

```
import pandas as pd

data = pd.read_csv('data/geysers.csv', sep=',')
r = np.array(data['waiting'])
```

In [10]:

```
# Estimate the mean waiting time and give a standard error for the estimate.
theta = r.mean()
se = r.std()

print("Estimated mean: %.3f" % theta)
print("Estimated SE: %.3f" % se)
```

```
Estimated mean: 70.897
Estimated SE: 13.570
```

In [11]:

```
# Also, give a 90% confidence interval for the mean waiting time.

import math
from scipy.stats import norm

z_90 = norm.ppf(.95)

print('90%% confidence interval: (%.3f, %.3f)' % ((theta - z_90 * se), (theta + z_90 * se)))
```

```
90% confidence interval: (48.576, 93.218)
```

In [12]:

```
# Now estimate the median time
```

```
median = np.median(r)

print("Estimated median time: %.3f" % median)
```

```
Estimated median time: 76.000
```

Exercise 8.5.9. 100 people are given a standard antibiotic to treat an infection and another 100 are given a new antibiotic. In the first group, 90 people recover; in the second group, 85 people recover. Let p_1 be the probability of recovery under the standard treatment, and let p_2 be the probability of recovery under the new treatment. We are interested in estimating $\theta = p_1 - p_2$. Provide an estimate, standard error, an 80% confidence interval and a 95% confidence interval for θ.

Solution. Let $X_1, \dots, X_100$ be indicator random variables (0 or 1) determining recovery on the first group, and $Y_1, \dots, Y_100$ indicating recovery on the second group. From the problem formulation, we can assume $X_i \sim \text{Bernoulli}(p_1)$ and $Y_i \sim \text{Bernoulli}(p_2)$.

If $\theta = p_1 - p_2$, then from exercise 8.5.2:

$$\hat{\theta} = \hat{p}_1 - \hat{p}_2$$

$$\text{se}(\hat{\theta}) = \sqrt{\hat{p}_1(1 - \hat{p}_1) + \hat{p}_2(1 - \hat{p}_2)}$$

In [13]:

```
import math

p_hat_1 = 0.9
p_hat_2 = 0.85

theta_hat = p_hat_1 - p_hat_2
se_theta_hat = math.sqrt(p_hat_1 * (1 - p_hat_1) + p_hat_2 * (1 - p_hat_2))

print('Estimated mean: %.3f' % theta_hat)
print('Estimated SE: %.3f'   % se_theta_hat)
```

```
Estimated mean: 0.050
Estimated SE: 0.466
```

In [14]:

```
from scipy.stats import norm

z_80 = norm.ppf(.9)
z_95 = norm.ppf(.975)

print('80%% confidence interval: (%.3f, %.3f)' % ((theta_hat - z_80 * se_theta_hat), (theta_hat + z_80 * se_theta_hat)))
print('95%% confidence interval: (%.3f, %.3f)' % ((theta_hat - z_95 * se_theta_hat), (theta_hat + z_95 * se_theta_hat)))
```

```
80% confidence interval: (-0.548, 0.648)
95% confidence interval: (-0.864, 0.964)
```

Exercise 8.5.10. In 1975, an experiment was conducted to see if cloud seeding produced rainfall. 26 clouds were seeded with silver nitrate and 26 were not. The decision to seed or not was made at random. Get the data from the provided link.

Let θ be the difference in the median precipitation from the two groups.

- Estimate θ.
- Estimate the standard error of the estimate and produce a 95% confidence interval.

In [15]:

```
import numpy as np
import pandas as pd
from tqdm import tqdm_notebook

data = pd.read_csv('data/cloud_seeding.csv', sep=',')
X = data['Seeded_Clouds']
Y = data['Unseeded_Clouds']
```

In [16]:

```
theta_hat = X.median() - Y.median()

print('Estimated mean: %.3f' % theta_hat)
```

```
Estimated mean: 177.400
```

In [17]:

```
# Using bootstrap (from chapter 9):

nx = len(X)
ny = len(Y)

B = 10000
t_boot = np.zeros(B)
for i in tqdm_notebook(range(B)):
    xx = X.sample(n=nx, replace=True)
    yy = Y.sample(n=ny, replace=True)
    t_boot[i] = xx.median() - yy.median()

se = np.array(t_boot).std()

print('Estimated SE: %.3f' % se)
```

```
Estimated SE: 62.363
```

In [18]:

```
# See example 9.5, page 135

from scipy.stats import norm

z_95 = norm.ppf(.975)

normal_conf = (theta_hat - z_95 * se, theta_hat + z_95 * se)
percentile_conf = (np.quantile(t_boot, .025), np.quantile(t_boot, .975))
pivotal_conf = (2*theta_hat - np.quantile(t_boot, 0.975), 2*theta_hat - np.quantile(t_boot, .025))

print('95%% confidence interval (Normal): \t %.3f, %.3f' % normal_conf)
print('95%% confidence interval (percentile): \t %.3f, %.3f' % percentile_conf)
print('95%% confidence interval (pivotal): \t %.3f, %.3f' % pivotal_conf)
```

```
95% confidence interval (Normal):        55.171, 299.629
95% confidence interval (percentile):    37.800, 262.560
95% confidence interval (pivotal):       92.240, 317.000
```