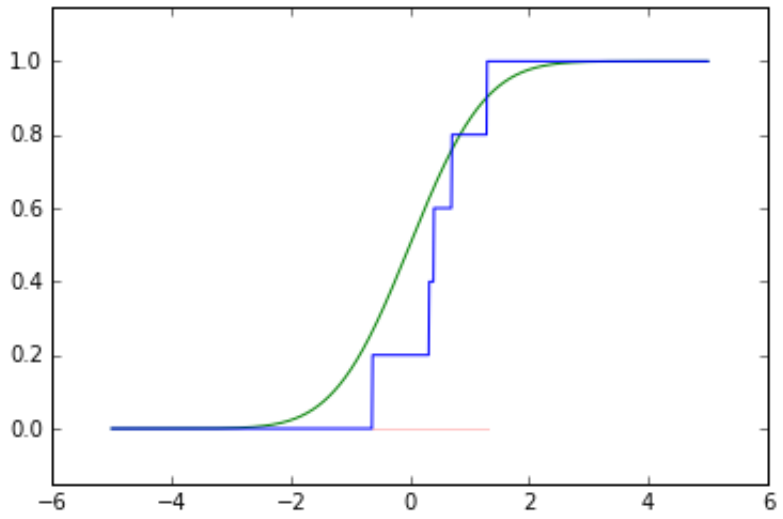
4.

In [89]:

```
import numpy as np
import scipy.stats as st
import matplotlib.pyplot as plt
%matplotlib inline

size_N = 10000

def emp (n):
    distribution_normal = st.norm(0, 1)
    sample = distribution_normal.rvs(size = size_N)
    sample = sample[:n]
    sample = np.sort(sample)
    spots = np.linspace(-5, 5, 1000)

    y = 1. * np.searchsorted(sample, spots) / n
    plt.plot(spots, distribution_normal.cdf(spots), 'g')
    plt.plot(spots, y, 'b')
    plt.plot(sample, np.zeros(n), 'r', alpha=0.2)
    plt.ylim([-0.15, 1.15])
    plt.show()
```

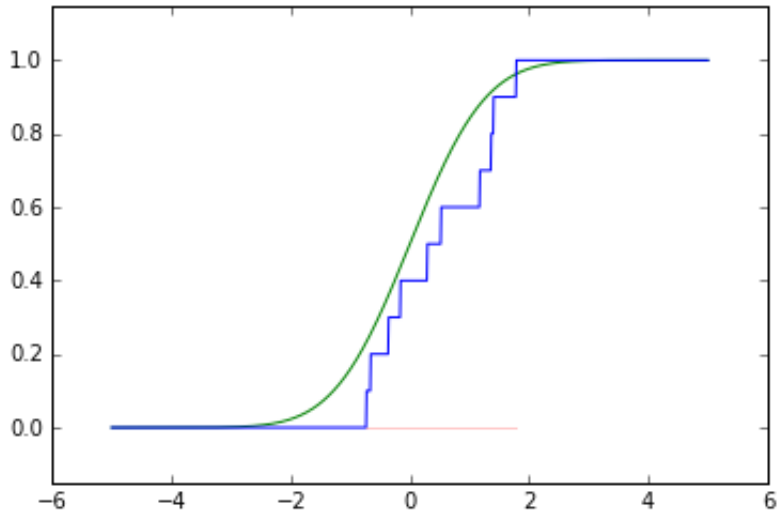
In [90]:

```
emp(5)
```



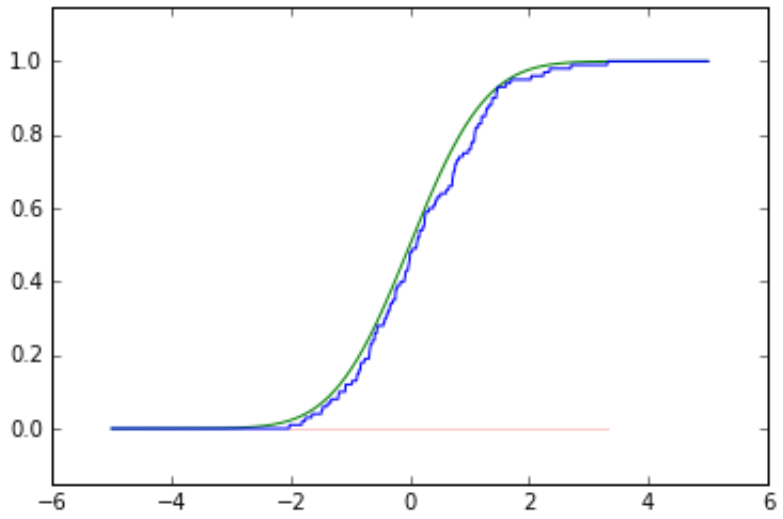
In [91]:

```
emp(10)
```



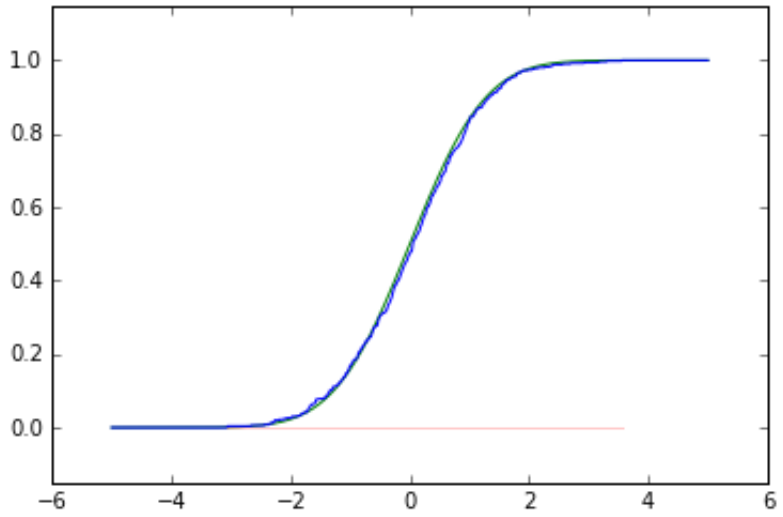
In [92]:

```
emp(100)
```



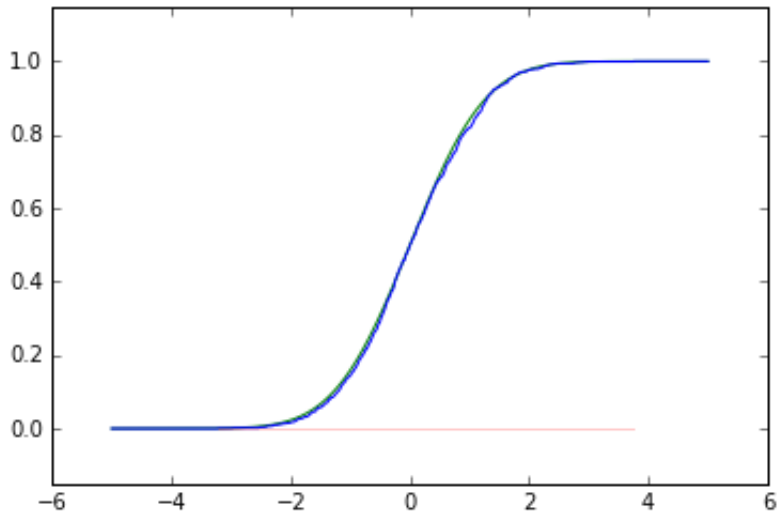
In [93]:

```
emp(500)
```



In [94]:

```
emp(1000)
```



In [111]:

```
distribution_normal = st.norm(0, 1)
sample = distribution_normal.rvs(size = size_N)

for k in range(2, size_N - 1):
    sorted_sample = np.sort(sample[:k])
    spots = np.linspace(-5, 5, 1000)
    y = np.arange(len(sorted_sample))/float(len(sorted_sample))
    sup[k] = max([abs(y[i] - distribution_normal.cdf(sorted_sample[i])) for i

plt.plot(sup, 'r')
plt.ylim([0, 0.15])
plt.xlim([0, 10000])
plt.show()
```