Finally, I calculated the cumulative strategy returns and the cumulative market returns and saved them in df. Then, I calculated the sharpe ratio to measure the performance. To get a clear understanding of this metric I plotted the performance to measure it.

```
df['strategy_cu_return']=0.
df['market_cu_return']=0.
df.iloc[-p_data:,df.columns.get_loc('strategy_cu_return')] \
       = np.nancumsum(df['str_ret'][-p_data:])
df.iloc[-p_data:,df.columns.get_loc('market_cu_return')] \
       = np.nancumsum(df['Return'][-p_data:])
Sharpe = (df['strategy_cu_return'][-1]-df['market_cu_return'][-1])\
          /np.nanstd(df['strategy_cu_return'][-p_data:])

plt.plot(df['strategy_cu_return'][-p_data:],color='g',label='Strategy Returns')
plt.plot(df['market_cu_return'][-p_data:],color='r',label='Market Returns')
plt.figtext(0.14,0.9,s='Sharpe ratio: %.2f'%Sharpe)
plt.legend(loc='best')
plt.show()
```

The final result looks like this.

After so much of code and effort, if the end result looks like this, then someone with no machine learning back ground would say that it is not worth it. I would agree for now. But, look at this line of code:

```
df= web.get_data_yahoo('IBM',start= '2000-01-01', end='2017-08-01')
```

I just changed the data from SPY to IBM. Then the result looks like this: