```
% Two Sample Independent t-test
% Andrew Brown
% GEEN 3853
% Fall 2020
```

INTRO

```
%For the 1-sample test, let’s test the hypothesis that the annual rate of
%return for the stock market over time is 8%.  For this data, let’s start
%1/1/1958 (the S&P adopted 500 stocks in 1957,
%https://www.investopedia.com/ask/answers/042415/what-average-annual-return-sp-500.asp)
%and go to 1/1/2018.  We will only want the 1/1 data for each year, and
%we need to calculate the % increase in the index for each year.
%The annual % increase is what we will test against 8%.

%The S&P 500 is a weighted and aggregated valuation of 500 companies that
%are representative of the broader economy.
```

Prepare workspace

```
clear all
close all
clc
```

LOAD DATA

```
%load as "table" for more versatility (readtable)

raw = readtable('sp500_data.csv');
```

```
Warning: Column headers from the file were modified to make them valid MATLAB identifiers before creating
 variable names for the table. The original column headers are saved in the VariableDescriptions property.
Set 'PreserveVariableNames' to true to use the original column headers as table variable names.
```

```

ds_full = raw;

%What is in this dataset? summary(ds), head(ds,3)
summary(ds_full)
```

```
Variables:

    Date: 1768×1 datetime

        Properties:
            Description:  Date
        Values:

            Min       1871-01-01
            Median    1944-08-16
            Max       2018-04-01

    SP500: 1768×1 double

        Properties:
            Description:  SP500
```

```
    Values:

        Min          2.73
        Median     16.335
        Max        2789.8
```

Dividend: 1768×1 double

```
    Properties:
        Description:  Dividend
    Values:

        Min             0.18
        Median          0.83
        Max             50
        NumMissing      1
```

Earnings: 1768×1 double

```
    Properties:
        Description:  Earnings
    Values:

        Min             0.16
        Median          1.325
        Max             109.88
        NumMissing      4
```

ConsumerPriceIndex: 1768×1 double

```
    Properties:
        Description:  Consumer Price Index
    Values:

        Min          6.28
        Median       18.2
        Max        249.84
```

LongInterestRate: 1768×1 double

```
    Properties:
        Description:  Long Interest Rate
    Values:

        Min          1.5
        Median      3.86
        Max        15.32
```

RealPrice: 1768×1 double

```
    Properties:
        Description:  Real Price
    Values:

        Min         67.63
        Median     253.06
        Max          2812
```

RealDividend: 1768×1 double

```
    Properties:
        Description:  Real Dividend
    Values:

        Min             4.98
```

```
        Median          12.72
        Max             50.06
        NumMissing      1

    RealEarnings: 1768×1 double

        Properties:
            Description:  Real Earnings
        Values:

            Min             4.19
            Median          20.445
            Max             111.36
            NumMissing      4

    PE10: 1768×1 double

        Properties:
            Description:  PE10
        Values:

            Min             4.78
            Median          16.17
            Max             44.2
            NumMissing      120
```

Simplify the dataset

```
ds = ds_full(:,{'Date', 'SP500'});
head(ds, 2)
```

ans = 2×2 table

	Date	SP500
1	1871-01-01	4.4400
2	1871-02-01	4.5000

```
%
```

Clean the data; fix data types (nominal or categorical)

We only want the data for the 1/1 of each year, starting 1/1/1958.

```
temp = (1940 >= year(ds.Date)) & (year(ds.Date)) >= 1930;
%ds(1045,:)
%ds58 = ds(1045:12:end,:)
dsDepressionMonth = ds(temp,:)
```

dsDepressionMonth = 132×2 table

	Date	SP500
1	1930-01-01	21.7100
2	1930-02-01	23.0700
3	1930-03-01	23.9400
4	1930-04-01	25.4600

	Date	SP500
5	1930-05-01	23.9400
6	1930-06-01	21.5200
7	1930-07-01	21.0600
8	1930-08-01	20.7900
9	1930-09-01	20.7800
10	1930-10-01	17.9200
11	1930-11-01	16.6200
12	1930-12-01	15.5100
13	1931-01-01	15.9800
14	1931-02-01	17.2000
15	1931-03-01	17.5300
16	1931-04-01	15.8600
17	1931-05-01	14.3300
18	1931-06-01	13.8700
19	1931-07-01	14.3300
20	1931-08-01	13.9000
21	1931-09-01	11.8300
22	1931-10-01	10.2500
23	1931-11-01	10.3900
24	1931-12-01	8.4400
25	1932-01-01	8.3000
26	1932-02-01	8.2300
27	1932-03-01	8.2600
28	1932-04-01	6.2800
29	1932-05-01	5.5100
30	1932-06-01	4.7700
31	1932-07-01	5.0100
32	1932-08-01	7.5300
33	1932-09-01	8.2600
34	1932-10-01	7.1200
35	1932-11-01	7.0500
36	1932-12-01	6.8200
37	1933-01-01	7.0900
38	1933-02-01	6.2500

	Date	SP500
39	1933-03-01	6.2300
40	1933-04-01	6.8900
41	1933-05-01	8.8700
42	1933-06-01	10.3900
43	1933-07-01	11.2300
44	1933-08-01	10.6700
45	1933-09-01	10.5800
46	1933-10-01	9.5500
47	1933-11-01	9.7800
48	1933-12-01	9.9700
49	1934-01-01	10.5400
50	1934-02-01	11.3200
51	1934-03-01	10.7400
52	1934-04-01	10.9200
53	1934-05-01	9.8100
54	1934-06-01	9.9400
55	1934-07-01	9.4700
56	1934-08-01	9.1000
57	1934-09-01	8.8800
58	1934-10-01	8.9500
59	1934-11-01	9.2000
60	1934-12-01	9.2600
61	1935-01-01	9.2600
62	1935-02-01	8.9800
63	1935-03-01	8.4100
64	1935-04-01	9.0400
65	1935-05-01	9.7500
66	1935-06-01	10.1200
67	1935-07-01	10.6500
68	1935-08-01	11.3700
69	1935-09-01	11.6100
70	1935-10-01	11.9200
71	1935-11-01	13.0400
72	1935-12-01	13.0400

	Date	SP500
73	1936-01-01	13.7600
74	1936-02-01	14.5500
75	1936-03-01	14.8600
76	1936-04-01	14.8800
77	1936-05-01	14.0900
78	1936-06-01	14.6900
79	1936-07-01	15.5600
80	1936-08-01	15.8700
81	1936-09-01	16.0500
82	1936-10-01	16.8900
83	1936-11-01	17.3600
84	1936-12-01	17.0600
85	1937-01-01	17.5900
86	1937-02-01	18.1100
87	1937-03-01	18.0900
88	1937-04-01	17.0100
89	1937-05-01	16.2500
90	1937-06-01	15.6400
91	1937-07-01	16.5700
92	1937-08-01	16.7400
93	1937-09-01	14.3700
94	1937-10-01	12.2800
95	1937-11-01	11.2000
96	1937-12-01	11.0200
97	1938-01-01	11.3100
98	1938-02-01	11.0400
99	1938-03-01	10.3100
100	1938-04-01	9.8900

⋮

```
dsDepression = dsDepressionMonth(1:12:end,:)
```

dsDepression = 11×2 table

	Date	SP500
1	1930-01-01	21.7100

	Date	SP500
2	1931-01-01	15.9800
3	1932-01-01	8.3000
4	1933-01-01	7.0900
5	1934-01-01	10.5400
6	1935-01-01	9.2600
7	1936-01-01	13.7600
8	1937-01-01	17.5900
9	1938-01-01	11.3100
10	1939-01-01	12.5000
11	1940-01-01	12.3000

```
temp = (2018 >= year(ds.Date)) & (year(ds.Date)) >= 2008;
%ds(1045,:)
%ds58 = ds(1045:12:end,:)
dsRecessionMonth = ds(temp,:)
```

dsRecessionMonth = 124×2 table

	Date	SP500
1	2008-01-01	1.3788e+03
2	2008-02-01	1.3549e+03
3	2008-03-01	1.3169e+03
4	2008-04-01	1.3705e+03
5	2008-05-01	1.4032e+03
6	2008-06-01	1.3413e+03
7	2008-07-01	1.2573e+03
8	2008-08-01	1.2815e+03
9	2008-09-01	1.2170e+03
10	2008-10-01	968.8000
11	2008-11-01	883.0400
12	2008-12-01	877.5600
13	2009-01-01	865.5800
14	2009-02-01	805.2300
15	2009-03-01	757.1300
16	2009-04-01	848.1500
17	2009-05-01	902.4100
18	2009-06-01	926.1200

	Date	SP500
19	2009-07-01	935.8200
20	2009-08-01	1.0097e+03
21	2009-09-01	1.0445e+03
22	2009-10-01	1.0677e+03
23	2009-11-01	1.0881e+03
24	2009-12-01	1.1104e+03
25	2010-01-01	1.1236e+03
26	2010-02-01	1.0892e+03
27	2010-03-01	1.1520e+03
28	2010-04-01	1.1973e+03
29	2010-05-01	1.1251e+03
30	2010-06-01	1.0834e+03
31	2010-07-01	1.0798e+03
32	2010-08-01	1.0873e+03
33	2010-09-01	1.1221e+03
34	2010-10-01	1.1716e+03
35	2010-11-01	1.1989e+03
36	2010-12-01	1.2415e+03
37	2011-01-01	1.2826e+03
38	2011-02-01	1.3211e+03
39	2011-03-01	1.3045e+03
40	2011-04-01	1.3315e+03
41	2011-05-01	1.3383e+03
42	2011-06-01	1.2873e+03
43	2011-07-01	1.3252e+03
44	2011-08-01	1.1853e+03
45	2011-09-01	1.1739e+03
46	2011-10-01	1.2072e+03
47	2011-11-01	1.2264e+03
48	2011-12-01	1.2433e+03
49	2012-01-01	1.3006e+03
50	2012-02-01	1.3525e+03
51	2012-03-01	1.3892e+03
52	2012-04-01	1.3864e+03

	Date	SP500
53	2012-05-01	1.3413e+03
54	2012-06-01	1.3235e+03
55	2012-07-01	1.3598e+03
56	2012-08-01	1.4035e+03
57	2012-09-01	1.4434e+03
58	2012-10-01	1.4378e+03
59	2012-11-01	1.3945e+03
60	2012-12-01	1.4223e+03
61	2013-01-01	1.4804e+03
62	2013-02-01	1.5123e+03
63	2013-03-01	1.5508e+03
64	2013-04-01	1.5707e+03
65	2013-05-01	1.6398e+03
66	2013-06-01	1.6188e+03
67	2013-07-01	1.6687e+03
68	2013-08-01	1.6701e+03
69	2013-09-01	1.6872e+03
70	2013-10-01	1.7200e+03
71	2013-11-01	1.7835e+03
72	2013-12-01	1.8078e+03
73	2014-01-01	1.8224e+03
74	2014-02-01	1.8170e+03
75	2014-03-01	1.8635e+03
76	2014-04-01	1.8643e+03
77	2014-05-01	1.8898e+03
78	2014-06-01	1.9471e+03
79	2014-07-01	1.9731e+03
80	2014-08-01	1.9615e+03
81	2014-09-01	1.9932e+03
82	2014-10-01	1.9373e+03
83	2014-11-01	2.0446e+03
84	2014-12-01	2.0543e+03
85	2015-01-01	2.0282e+03
86	2015-02-01	2.0822e+03

	Date	SP500
87	2015-03-01	2.0800e+03
88	2015-04-01	2.0949e+03
89	2015-05-01	2.1119e+03
90	2015-06-01	2.0993e+03
91	2015-07-01	2.0941e+03
92	2015-08-01	2.0399e+03
93	2015-09-01	1.9444e+03
94	2015-10-01	2.0248e+03
95	2015-11-01	2.0806e+03
96	2015-12-01	2.0541e+03
97	2016-01-01	1.9186e+03
98	2016-02-01	1.9044e+03
99	2016-03-01	2.0220e+03
100	2016-04-01	2.0755e+03

⋮

```
dsRecession = dsRecessionMonth(1:12:end,:)
```

dsRecession = 11×2 table

	Date	SP500
1	2008-01-01	1.3788e+03
2	2009-01-01	865.5800
3	2010-01-01	1.1236e+03
4	2011-01-01	1.2826e+03
5	2012-01-01	1.3006e+03
6	2013-01-01	1.4804e+03
7	2014-01-01	1.8224e+03
8	2015-01-01	2.0282e+03
9	2016-01-01	1.9186e+03
10	2017-01-01	2.2751e+03
11	2018-01-01	2.7898e+03

Calculate annual rate of return

```
    %Neglecting inflation and dividends
dsDepression.ARR(1) = nan;
for ii =2:height(dsDepression)
```

```
dsDepression.ARR(ii) = (dsDepression.SP500(ii)-dsDepression.SP500(ii-1))/dsDepression.SP500(ii-
end

summary(dsDepression)
```

```
Variables:

    Date: 11×1 datetime

        Properties:
            Description:  Date
        Values:

            Min       1930-01-01
            Median    1935-01-01
            Max       1940-01-01

    SP500: 11×1 double

        Properties:
            Description:  SP500
        Values:

            Min         7.09
            Median      12.3
            Max        21.71

    ARR: 11×1 double

        Values:

            Min              -0.4806
            Median           -0.068721
            Max              0.4866
            NumMissing       1
```

```
%nanmean(ds58.ARR)

 %Neglecting inflation and dividends
dsRecession.ARR(1) = nan;
for ii =2:height(dsRecession)
dsRecession.ARR(ii) = (dsRecession.SP500(ii)-dsRecession.SP500(ii-1))/dsRecession.SP500(ii-1);
end

summary(dsRecession)
```

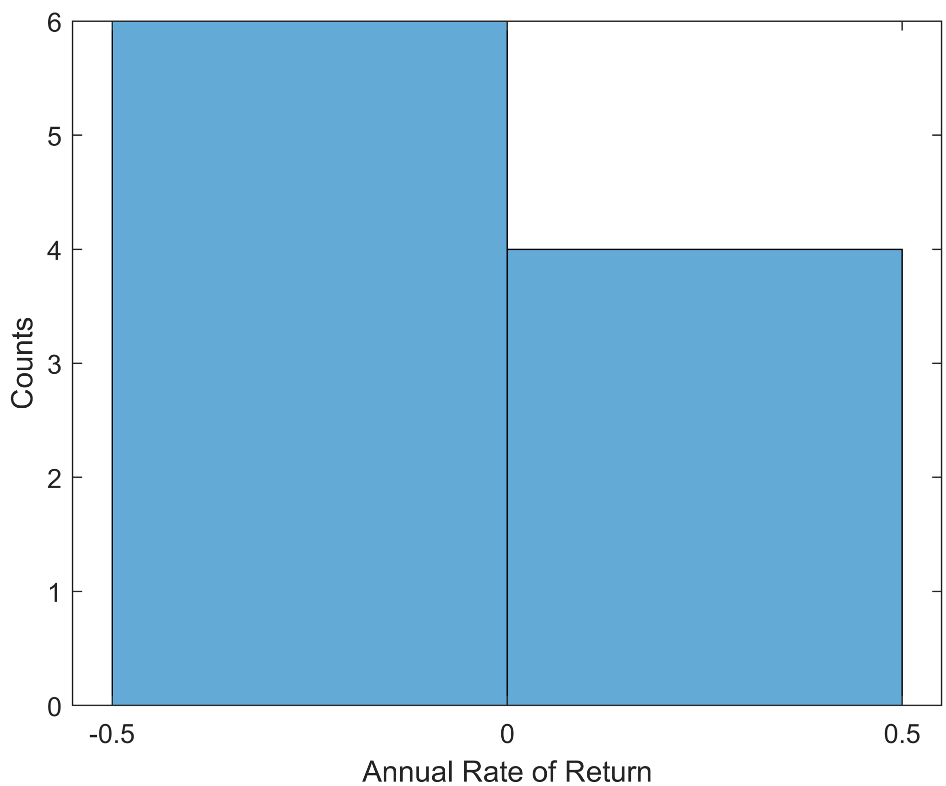
```
Variables:

    Date: 11×1 datetime

        Properties:
            Description:  Date
        Values:

            Min       2008-01-01
            Median    2013-01-01
            Max       2018-01-01

    SP500: 11×1 double

        Properties:
            Description:  SP500
```

```
    Values:

        Min        865.58
        Median     1480.4
        Max        2789.8

ARR: 11×1 double

    Values:

        Min             -0.3722
        Median          0.1399
        Max             0.29807
        NumMissing      1
```

```
%nanmean(ds58.ARR)
```

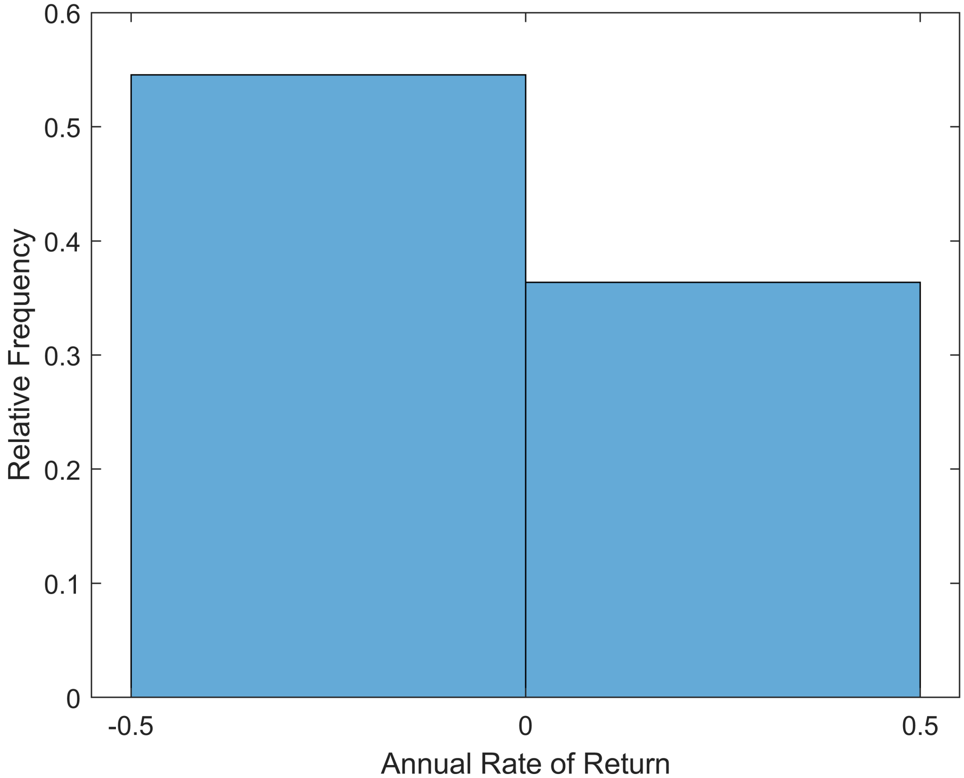
DESCRIPTIVE STATS - VISUAL

```
% %% Descriptive stats: Continuous
% Plot (histogram)
figure
h = histogram(dsDepression.ARR);
xlabel('Annual Rate of Return')
ylabel('Counts')
```

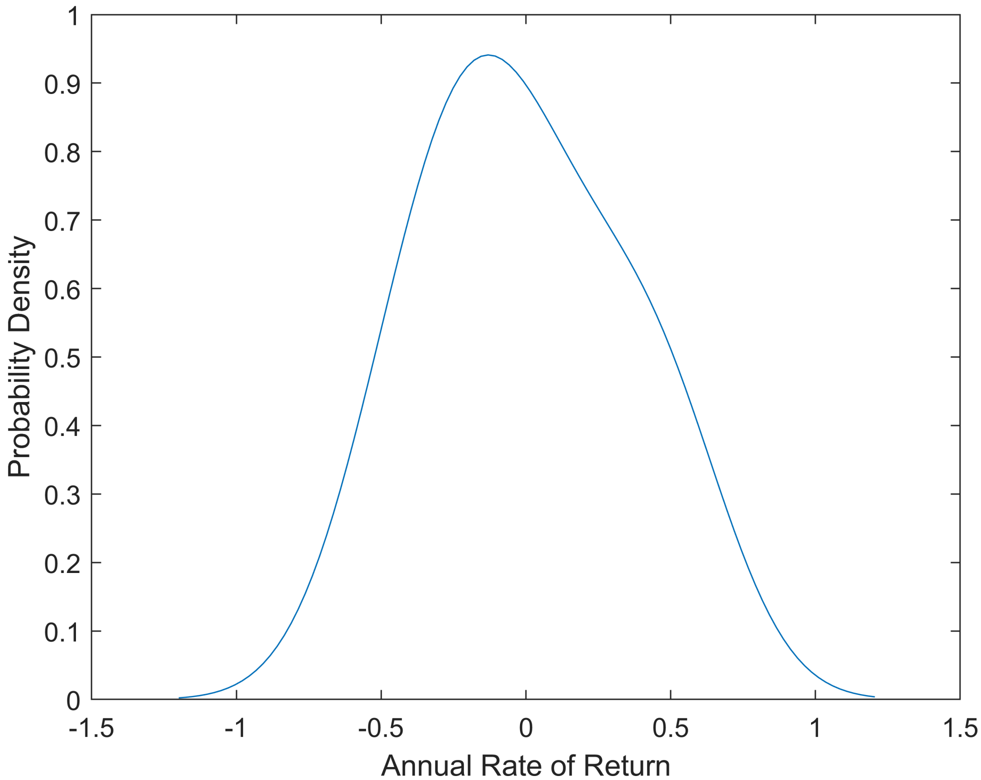



```
%Relative Frequency
figure
h = histogram(dsDepression.ARR, 'Normalization', 'probability');
xlabel('Annual Rate of Return')
```

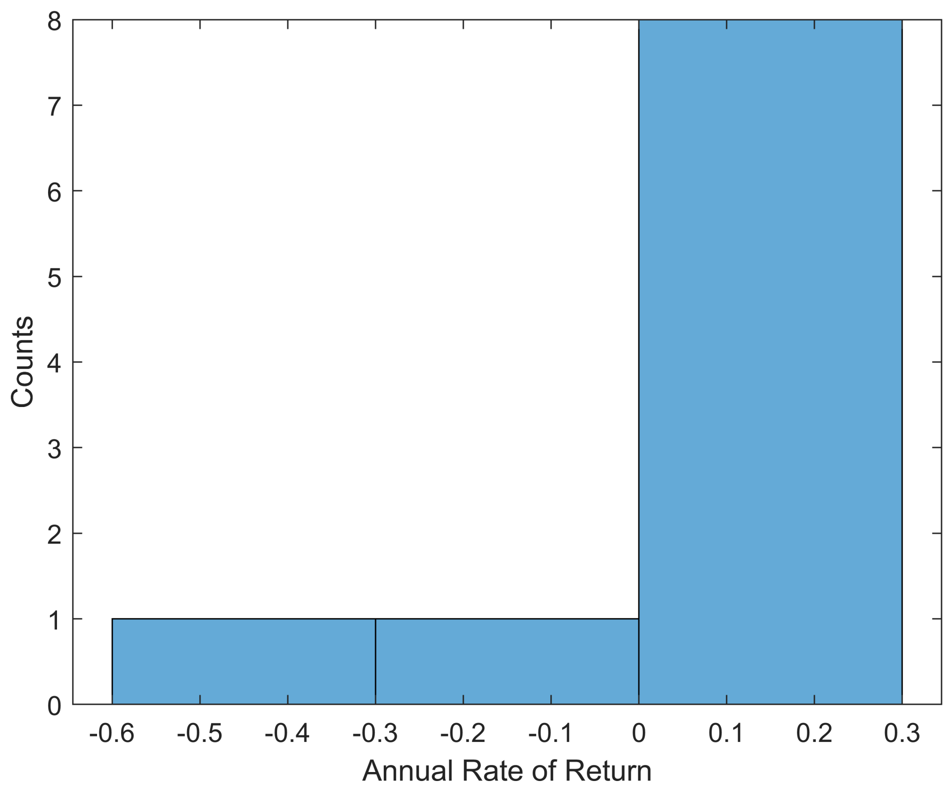
```
ylabel('Relative Frequency')
```



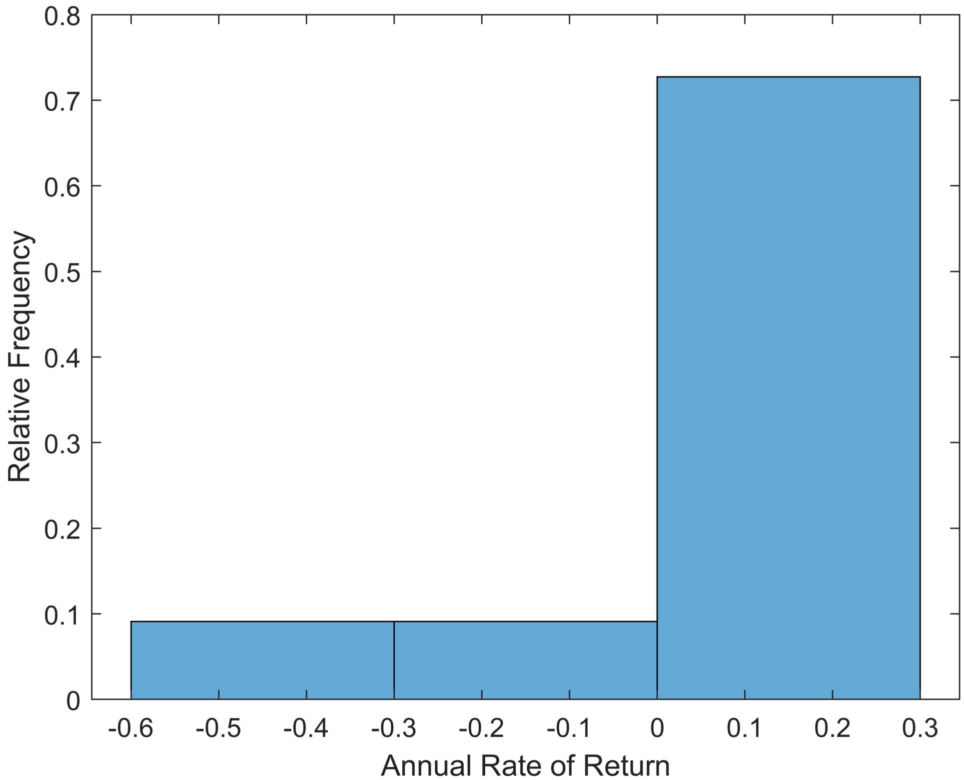

```
%Probability Distribution Function based on this sample:
figure
ksdensity(dsDepression.ARR);
xlabel('Annual Rate of Return')
ylabel('Probability Density')
```



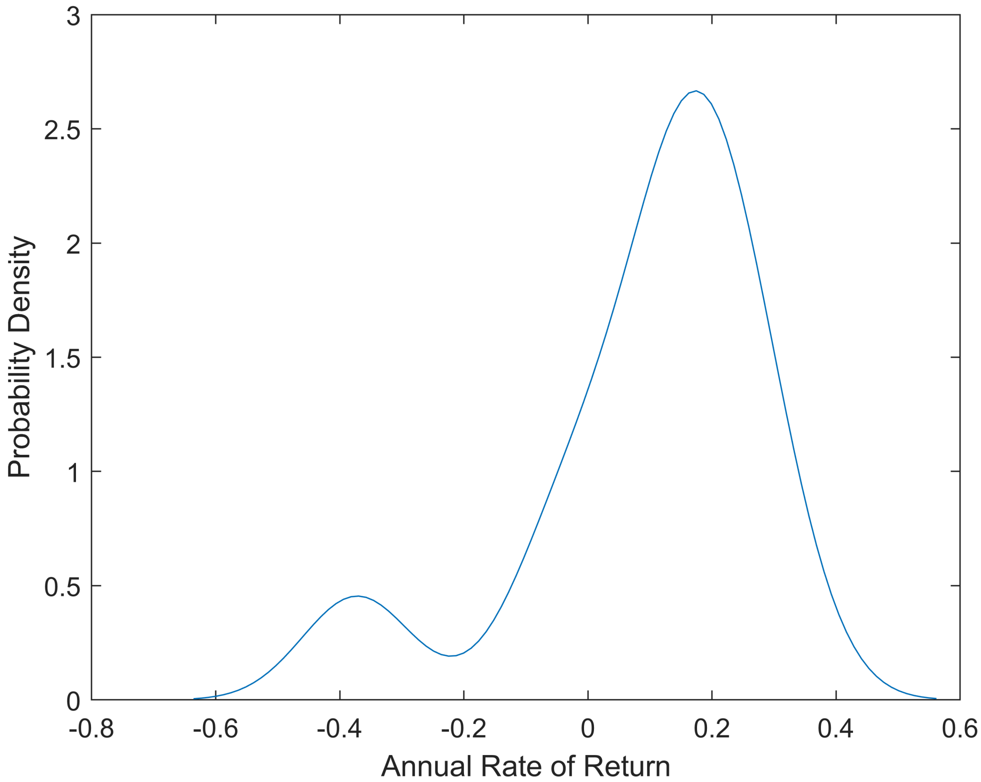

```
%QUESTION: Why are some values greater than one?!

% %% Descriptive stats: Continuous
% Plot (histogram)
figure
h = histogram(dsRecession.ARR);
xlabel('Annual Rate of Return')
ylabel('Counts')
```




```
%Relative Frequency
figure
h = histogram(dsRecession.ARR, 'Normalization', 'probability');
xlabel('Annual Rate of Return')
ylabel('Relative Frequency')
```




```
%Probability Distribution Function based on this sample:
figure
ksdensity(dsRecession.ARR);
xlabel('Annual Rate of Return')
ylabel('Probability Density')
```




```
%QUESTION: Why are some values greater than one?!
```

DESCRIPTIVE STATS - NUMERIC

Mean (mean)

```
mnD = nanmean(dsDepression.ARR)
```

```
mnD = -0.0029
```

```
% Median (median)
medD = nanmedian(dsDepression.ARR)
```

```
medD = -0.0687
```

```
% Mode (mode)
modD = mode(dsDepression.ARR)
```

```
modD = -0.4806
```

```
%When multiple values occur equally frequently, mode returns the smallest #

% Standard Deviation (std)
sdD = nanstd(dsDepression.ARR)
```

```
sdD = 0.3376
```

```
% Variance (var)
```

```
varD = nanvar(dsDepression.ARR)
```

```
varD = 0.1140
```

```
% Range (max() - min() or range)
rgD = range(dsDepression.ARR)
```

```
rgD = 0.9672
```

```
% Mean (mean)
mnR = nanmean(dsRecession.ARR)
```

```
mnR = 0.0922
```

```
% Median (median)
medR = nanmedian(dsRecession.ARR)
```

```
medR = 0.1399
```

```
% Mode (mode)
modR = mode(dsRecession.ARR)
```

```
modR = -0.3722
```

```
%When multiple values occur equally frequently, mode returns the smallest #

% Standard Deviation (std)
sdR = nanstd(dsRecession.ARR)
```

```
sdR = 0.1934
```

```
% Variance (var)
varR = nanvar(dsRecession.ARR)
```

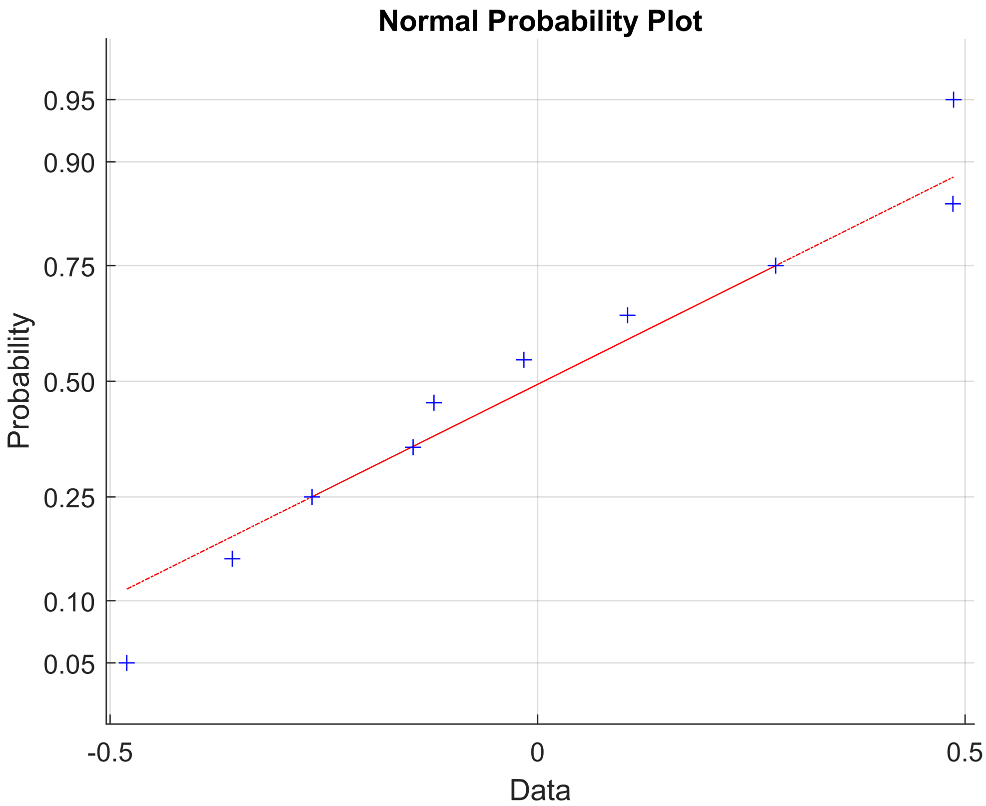
```
varR = 0.0374
```

```
% Range (max() - min() or range)
rgR = range(dsRecession.ARR)
```

```
rgR = 0.6703
```

TEST OF NORMALITY

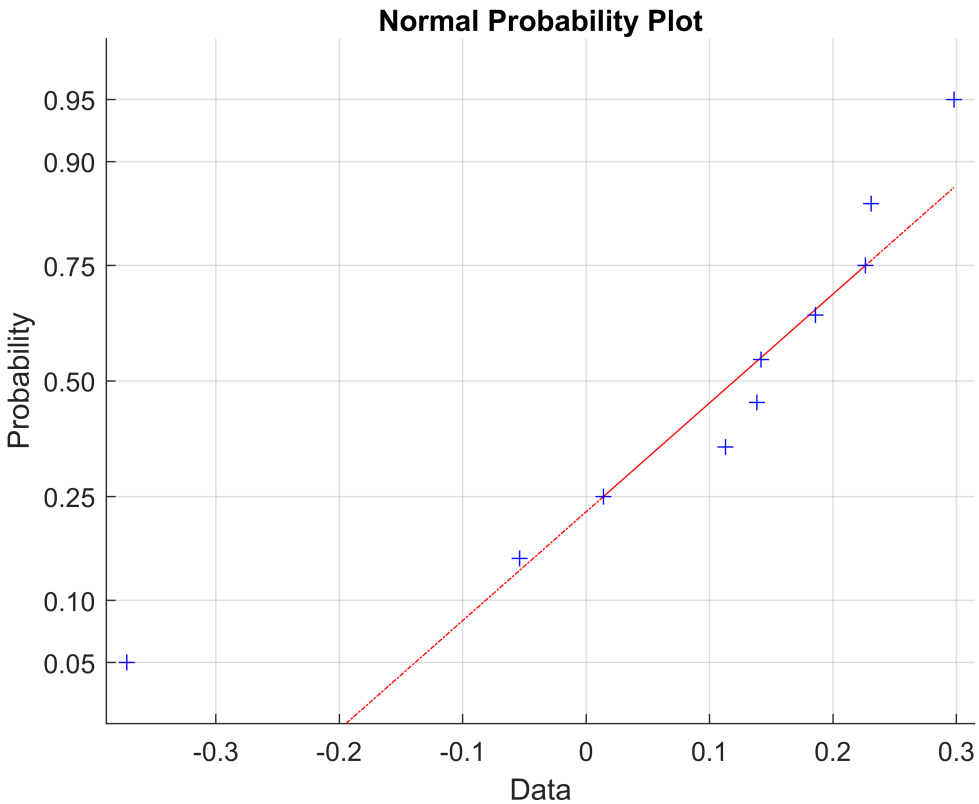
```
%Plot the data. The "normplot" function makes a normal probabiltiy plot of
%the data.  The purpose of this plot is to graphically assess whethere the
%data could come from a normal distribution.  The data is assumed to be
%normally distributed if the data points appear to fall on the linear
%regression line.

normplot (dsDepression.ARR)
```




```
%Next we want to use the "adtest" function to run the Anderson-Darling
%goodness-of-fit test. Often we reject the null hypothesis (that data is
%normally distributed) with significance level of 0.05.
%The Null hypothesis can be rejected if
%ADTEST > Critical Value (CV). For more information, remember that you can
%always use the "help" function in Matlab.

[H,P,ADSTAT,CV]=adtest(dsDepression.ARR, 'Alpha', 0.05)
```

```
H = logical
   0
P = 0.7498
ADSTAT = 0.2365
CV = 0.6857
```

```
%Plot the data. The "normplot" function makes a normal probabiltiy plot of
%the data.  The purpose of this plot is to graphically assess whethere the
%data could come from a normal distribution.  The data is assumed to be
%normally distributed if the data points appear to fall on the linear
%regression line.

normplot (dsRecession.ARR)
```




```
%Next we want to use the "adtest" function to run the Anderson-Darling
%goodness-of-fit test. Often we reject the null hypothesis (that data is
%normally distributed) with significance level of 0.05.
%The Null hypothesis can be rejected if
%ADTEST > Critical Value (CV). For more information, remember that you can
%always use the "help" function in Matlab.

[H,P,ADSTAT,CV]=adtest(dsRecession.ARR, 'Alpha', 0.05)
```

```
H = logical
   0
P = 0.0672
ADSTAT = 0.6400
CV = 0.6857
```

TEST OF VARIANCE

```
depressionVariance = (std(dsDepression.ARR))^2
```

```
depressionVariance = NaN
```

```
recessionVariance = (std(dsRecession.ARR))^2
```

```
recessionVariance = NaN
```

TEST OF LOCATION

```
%Choose the Type 1 error rate. Remember, "alpha" is a function in Matlab,
%so a1 is chosen for this variable name.

alpha1 = 0.05;

%------------------------One Sample Test------------------------------%
mu0 =  0.08;  %enter specified mean parameter/ given mean value (scalar)

%This function runs a t-test on data that is normally distributed.
%Reject the null hypothesis when H = 1, accept null when H = 0
%P is the p-value.  CI is the confidence interval for the true population
%mean. It is the (1-alpha)*100 percent CI.
%"STATS" will return the t-statistic, degrees of freedom,
%and the estimated population standard deviation
%(our best guess of the pop. stand. dev. is the sample standard deviation).
%Finally, the 'tail' must be specified as a two-tailed test ('both'),
%or a one-tailed test ('left' or 'right'). For more information, use help.

[H,P,CI,STATS] = ttest(dsDepression.ARR,mu0,'alpha',alpha1,'tail','both')
```

```
H = 0
P = 0.4575
CI = 2×1
   -0.2444
    0.2386
STATS = struct with fields:
    tstat: -0.7762
       df: 9
       sd: 0.3376
```

```
%---------------------------------------------------------------------------%

%Choose the Type 1 error rate. Remember, "alpha" is a function in Matlab,
%so a1 is chosen for this variable name.

alpha1 = 0.05;

%------------------------One Sample Test------------------------------%
mu0 =  0.08;  %enter specified mean parameter/ given mean value (scalar)

%This function runs a t-test on data that is normally distributed.
%Reject the null hypothesis when H = 1, accept null when H = 0
%P is the p-value.  CI is the confidence interval for the true population
%mean. It is the (1-alpha)*100 percent CI.
%"STATS" will return the t-statistic, degrees of freedom,
%and the estimated population standard deviation
%(our best guess of the pop. stand. dev. is the sample standard deviation).
%Finally, the 'tail' must be specified as a two-tailed test ('both'),
%or a one-tailed test ('left' or 'right'). For more information, use help.

[H,P,CI,STATS] = ttest(dsRecession.ARR,mu0,'alpha',alpha1,'tail','both')
```

```
H = 0
P = 0.8468
CI = 2×1
   -0.0462
```

```
    0.2305
STATS = struct with fields:
    tstat: 0.1989
       df: 9
       sd: 0.1934
```

```
%-------------------------------------------------------------------------%
```

Proof of CI

```
t_cf = tinv(1 - 0.025, length(dsDepression.ARR)-1)
```

```
t_cf = 2.2281
```

```
int = t_cf*nanstd(dsDepression.ARR)/sqrt(length(dsDepression.ARR))
```

```
int = 0.2268
```

```
mn = nanmean(dsDepression.ARR)
```

```
mn = -0.0029
```

```
[mn - int, mn + int]
```

```
ans = 1×2
   -0.2297    0.2239
```

```

% We will go over confidence intervals later.

t_cf = tinv(1 - 0.025, length(dsRecession.ARR)-1)
```

```
t_cf = 2.2281
```

```
int = t_cf*nanstd(dsRecession.ARR)/sqrt(length(dsRecession.ARR))
```

```
int = 0.1299
```

```
mn = nanmean(dsRecession.ARR)
```

```
mn = 0.0922
```

```
[mn - int, mn + int]
```

```
ans = 1×2
   -0.0377    0.2221
```

```

% We will go over confidence intervals later.
```

Effect Size

```
%Coming soon
%-------------------------------------------------------------------------%
```

Conclusion

For the 10 year period after the start of the Great Depression, the Average Rate of Return for the S&P90 was -0.29%. For the 10 year period after the start of the Great Recession, the Average Rate of Return for the S&P500 was +9.22%.