DANIELS
COLLEGE OF BUSINESS
UNIVERSITY of DENVER

Python for Data Analytics

ADV STAT Module: Lesson 1

Training Manual

ADVSTAT: Lesson 1

Lecture for ADVSTAT Lesson 1

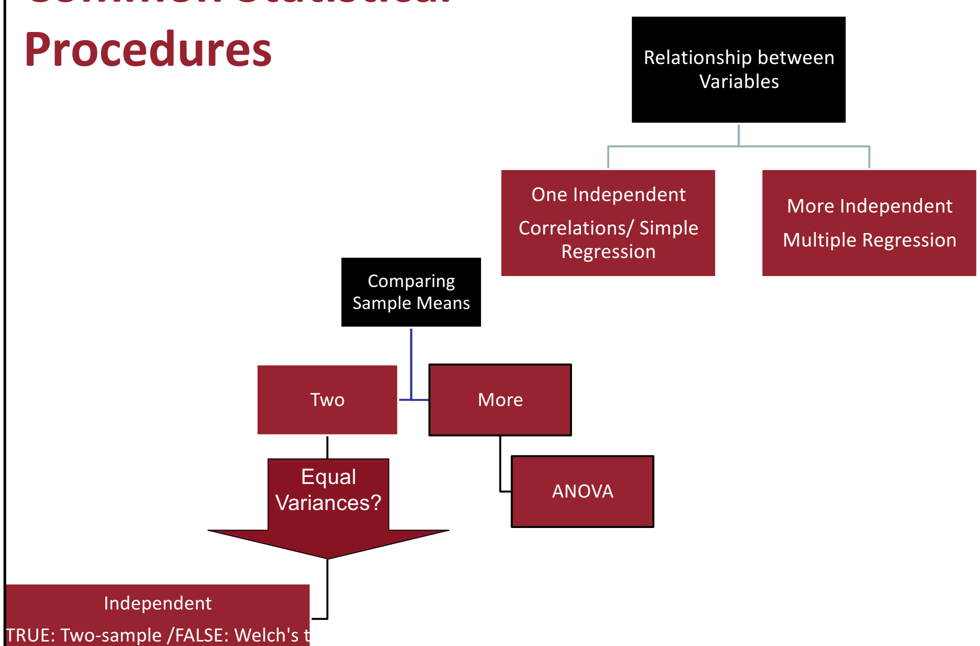
Mini Assignment ADVSTAT L1

Objectives Lesson 1

- Lesson 1
  - Statistical Testing
    - Comparing Means
      - Two-sample t test
        - equal variances test
      - ANOVA
    - Relationships between Variables
      - Simple Linear Regression

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

Common Statistical Procedures



How Hypothesis Testing Works

1. Ho: Null Hypothesis: Statement assumed True

   Ha: Alternate hypothesis

2. Alpha (α)= .05 (often we use .05 – could be different)

   This is the significance level/Type I Error

   The chance of Rejecting Ho but Ho was True!

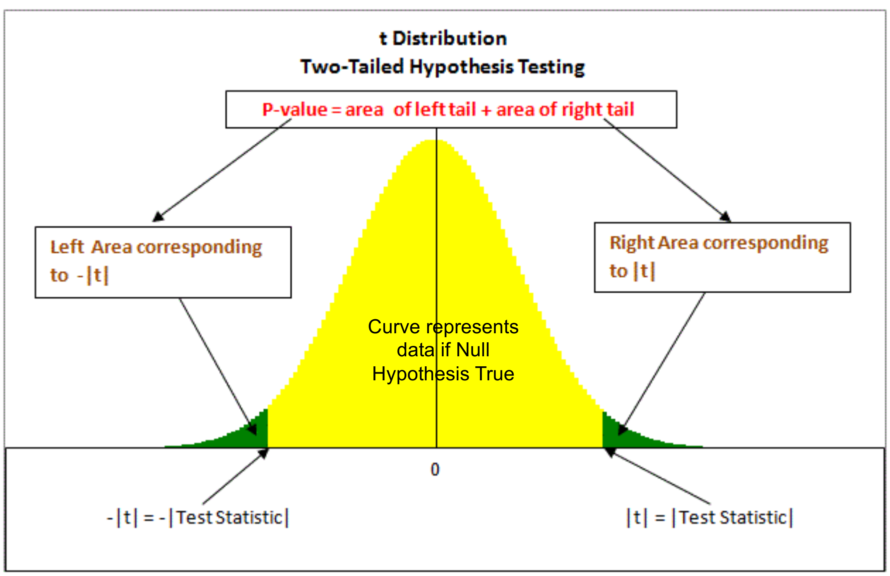
3. Find Test Statistic and compute p-value (see next slide)
4. Make Decision: Reject Ho if p-value < α
5. State Conclusion in terms of actual problem

   (i.e., the Sales mean for Chocolate orders is not the same as the Sales mean for Vanilla orders)

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

Hypothesis Testing – p-value

p-value represents probability that Ho is True given sample data



http://simulation-math.com/_ElementaryStatistics/InferenceTwoMeansC4.cshtml

Suppose we have the following, what is the Decision?

Ho: Vanilla and Chocolate Sales means are the same
p-value from Test Statistic: 0.450
Alpha α = .05

1. Reject Ho
2. Fail to Reject / Do Not Reject Ho

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

Suppose we have the following, what is the Conclusion?

Ho: Vanilla and Chocolate Sales means are the same
p-value from Test Statistic: 0.043
Alpha α = .05

1. Vanilla and Chocolate Sales means are the same
2. Vanilla and Chocolate Sales means are different

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

Comparing Means

As a motivating example, let's look at some Ice Cream sales data that has the age of the customer, their type (Adult, Child, Teenager), the flavor they bought (Chocolate or Vanilla), and the sales amount.

```
import os
os.chdir('/Users/Kellie/ ... /Lesson 1') #Mac
#os.chdir('C:\\Users\\Kellie\\ ... \\Lesson 1') #Windows

ICData = pd.read_excel('ADVSTATL1IceCreamData.xlsx')
ICData = ICData.drop('Customer', axis=1)
ICData.head()
```

```
        Type     Flavor  Age  Sales
0      Adult  Chocolate   45   4.25
1      Child    Vanilla    5   2.90
2   Teenager  Chocolate   14   3.10
3      Adult    Vanilla   23   3.25
4      Adult  Chocolate   47   4.10
```

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

Comparing Means – Two Sample t

Suppose we want to compare the means of the Flavors. Since we have 2 values, we can do a 2-sample t test assuming independent samples.

Let's first subset the data into a Choc and Vanilla DataFrame.

```
Choc=ICData[ICData['Flavor']=='Chocolate']
Choc.head()
```

```
       Type     Flavor  Age  Sales
0     Adult  Chocolate   45   4.25
2  Teenager  Chocolate   14   3.10
4     Adult  Chocolate   47   4.10
5  Teenager  Chocolate   16   4.10
6     Adult  Chocolate   41   3.50
```

```
Van=ICData[ICData['Flavor']=='Vanilla']
Van.head()
```

```
Type     Flavor  Age  Sales
1       Child  Vanilla    5   2.90
3       Adult  Vanilla   23   3.25
7       Child  Vanilla    4   3.00
9       Child  Vanilla    6   2.50
11   Teenager  Vanilla   11   3.00
```

Comparing Means – Two Sample t

We'll use the **stats** functions from **scipy** for most of our statistical tests. (We will also use some functions from **statsmodel** to round out our analyses.)

```
from scipy import stats
```

When running the test we need to know:

Can we assume the populations have equal variances?

- TRUE: The default t test assumes that there are equal population variances and performs a standard independent 2 sample test.
- FALSE: Otherwise, perform Welch's t-test which does not assume equal population variances.

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

Testing for Equal Variances

Next we'll perform a test for equal variances. For the test of variances, we will use the Bartlett test and alpha = .05. The null (assumed) hypothesis is:

Ho: The variances are the same.

Syntax: **stats.bartlett(data1,data2)** returns t test statistic and p-value.

```
alpha = .05
tvar, p_valvar = stats.bartlett(Choc['Sales'],Van['Sales'])
print("This is a test of equal variances with Ho: The variances
       are equal")
print("The t test statistic is " + str(round(tvar,3)) + " and the
       p-value is " + str(round(p_valvar,4)))
if p_valvar < alpha:
    print("Conclusion: Reject Ho: The variances are not equal")
    tEqVar=False
    ttype='Welch (unequal variances) Two-Sample t test'
else:
    print("Conclusion: Fail to Reject Ho: We can't reject that
           the variances are the same")
    tEqVar=True
    ttype='Two-Sample t test (assuming equal variances)'
```

```
This is a test of equal variances with Ho: The variances are equal
The t test statistic is 0.155 and the p-value is 0.6938
Conclusion: Fail to Reject Ho: We can't reject that the variances
are the same
```

Comparing Means – Two Sample t

Now that we know that we think the variances are the same, we can use the **equal_var=True** option on the t test with alpha = .05.

For the test of means, we will use the two sample t test:

Ho: The means are the same.

Syntax: **stats.ttest_ind(data1,data2,equal_var)** returns t test statistic and p-value

from Bartlett's test of variances:

```
tEqVar=True
ttype='Two-Sample t test (assuming equal variances)'
```

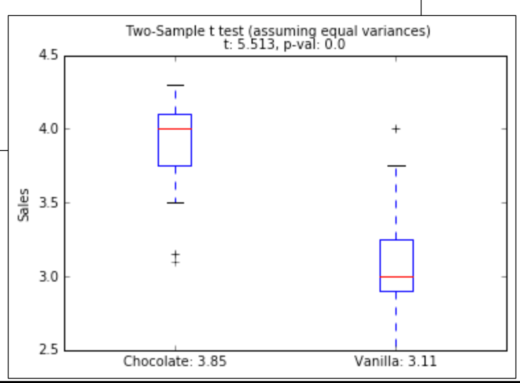
```
alpha = .05
tmean, p_valmean =
stats.ttest_ind(Choc['Sales'],Van['Sales'],equal_var=tEqVar)
print("This is a " + ttype + " of equal means with Ho: The group
      means are equal")
print("The t test statistic is " + str(round(tmean,3)) + " and
       the p-value is " + str(round(p_valmean,4)))
if p_valmean < alpha:
    print("Conclusion: Reject Ho: The means are not equal")
else:
    print("Conclusion: Fail to Reject Ho: We can't reject that
           the means are the same")
```

```
This is a Two-Sample t test (assuming equal variances) of
equal means with Ho: The group means are equal
The t test statistic is 5.513 and the p-value is 0.0
Conclusion: Reject Ho: The means are not equal
```

Comparing Two Means – Box Plot

A Box Plot is one way to compare group means.

```
# Create the boxplot
y=[Choc['Sales'],Van['Sales']]
plt.boxplot(y)
plt.title('t: ' + str(round(tmean,3)) + ', p-val: ' +
str(round(p_valmean,4)),size=10)
plt.suptitle(ttype,size=10)
plt.xticks(np.arange(1,3),['Chocolate: ' +
 str(round(Choc['Sales'].mean(),2)),
 'Vanilla: ' +
 str(round(Van['Sales'].mean(),2))])
plt.ylabel('Sales')
plt.savefig('ttest.png',
       bbox_inches='tight')
plt.show()
```



**Final Conclusion:
Chocolate Mean Sales are not the same as Vanilla Sales. From the graph, we can see they are higher.**

Comparing Means – ANOVA

If we have more than 2 means, then we can calculate a One-Way ANOVA test. Our data for Type has 3 categories: Adult, Child, and Teenager. Let's compare their means.

```
Adult = ICData[ICData['Type']=='Adult']
Adult[:3]
```

```
    Type     Flavor  Age  Sales
0  Adult  Chocolate   45   4.25
3  Adult    Vanilla   23   3.25
4  Adult  Chocolate   47   4.10
```

```
Child = ICData[ICData['Type']=='Child']
Child[:3]
```

```
 Type    Flavor  Age  Sales
1  Child  Vanilla    5    2.9
7  Child  Vanilla    4    3.0
9  Child  Vanilla    6    2.5
```

```
Teen = ICData[ICData['Type']=='Teenager']
Teen[:3]
```

```
Type      Flavor  Age  Sales
2  Teenager  Chocolate   14    3.1
5  Teenager  Chocolate   16    4.1
8  Teenager  Chocolate   17    4.0
```

Comparing Means – ANOVA

For the oneway ANOVA test of more than 2 means, we will use:

Ho: The group means are the same / Ha: At least one group differs

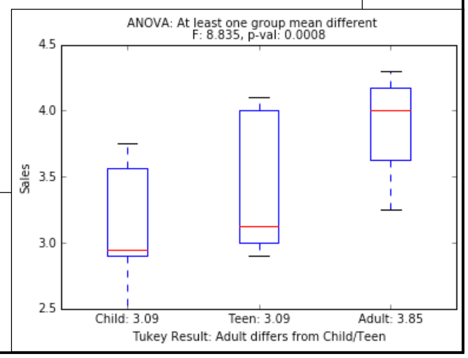
Syntax: **`stats.f_oneway (data1,data2,data3,…)`** returns f test statistic and p-value

```
alpha = .05
f, p_val = stats.f_oneway(Adult['Sales'], Child['Sales'],
Teen['Sales'])
print("This is a test of equal means with Ho: The means of all
groups are equal/Ha: At least one group mean is different")
print("The F test statistic is " + str(round(f,3)) + " and the p-
value is " + str(round(p_val,4)))
if p_val < alpha:
    print("Conclusion: Reject Ho: At least one group mean is
different")
    ANOVAtype = "ANOVA: At least one group mean different"
else:
    print("Conclusion: Fail to Reject Ho: We can't reject that the
means are the same")
    ANOVAtype = "ANOVA: Group Means are the same"
```

```
This is a test of equal means with Ho: The means of all groups
are equal/Ha: At least one group mean is different
The F test statistic is 8.835 and the p-value is 0.0008
Conclusion: Reject Ho: At least one group mean is different
```

Comparing Means – ANOVA – Box Plot

```
y=[Child['Sales'],Teen['Sales'],Adult['Sales'],]
plt.boxplot(y)
plt.title('F: ' + str(round(f,3)) + ', p-val: ' +
str(round(p_val,4)),size=10)
plt.suptitle(ANOVAtype,size=10)
plt.savefig('ANOVA.png', bbox_inches='tight')
plt.xticks(np.arange(1,4),['Child: '+
  str(round(Child['Sales'].mean(),2)),
                           'Teen: '+
  str(round(Child['Sales'].mean(),2)),
                           'Adult: '+
  str(round(Adult['Sales'].mean(),2))])
#Next slide Tukey Multiple Comparisons
plt.xlabel('Tukey Result: Adult differs
            from Child/Teen')
plt.ylabel('Sales')
plt.savefig('ANOVA.png',
             bbox_inches='tight')
plt.show()
```



Final Conclusion:
Child, Teen, and Adult mean Sales are not all the same – at least one differs from the others.

Comparing Means – ANOVA

If we conclude there is a difference, we need to perform a multiple comparisons test (controlling for overall alpha) to find the differences. We will use the Tukey pairwise test from **`statsmodels`**.

```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
# Data (endogenous/response variable)

tukey = pairwise_tukeyhsd(endog=ICData['Sales'],
                          groups=ICData['Type'], alpha=0.05)
print('Ho: The group means are equal')
print(tukey.summary() )
```

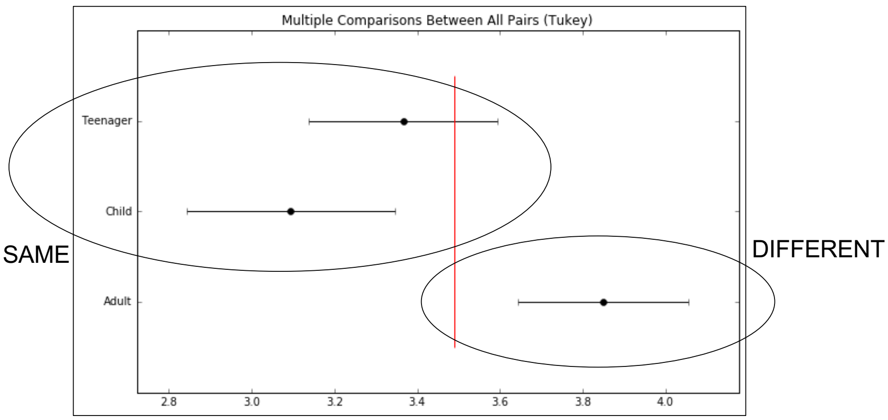
```
Ho: The group means are equal
Multiple Comparison of Means - Tukey HSD,FWER=0.05
===============================================
group1  group2  meandiff  lower   upper  reject
-----------------------------------------------
Adult   Child    -0.755  -1.2119 -0.2981  True
Adult  Teenager -0.4833  -0.9168 -0.0499  True
Child  Teenager  0.2717  -0.2075  0.7508 False
-----------------------------------------------
```

Final Conclusion:
Child and Teen mean Sales are the same.

Comparing Means – ANOVA - Graph

`statsmodels` also includes a plot that allows you to compare the confidence intervals for each of the groups. We also have added a vertical line at the overall mean.

```
tukey.plot_simultaneous()
# Plot group confidence intervals
plt.vlines(x=ICData['Sales'].mean(),ymin=-0.5,ymax=2.5,
              color="red")
```



Regression - Simple

Finally, if we want to examine the relationship between 2 variables, we can use regression. The Y dependent/response variable is what we want to predict and the x independent variable(s) is what we predict with.

Ho: X does not help to predict Y / slope is 0

Syntax: **`stats.linregress(x,y)`** returns slope, intercept, r_value, p-value, and standard error

```
xvar='Age'
yvar='Sales'
x=ICData[xvar]
y=ICData[yvar]
```

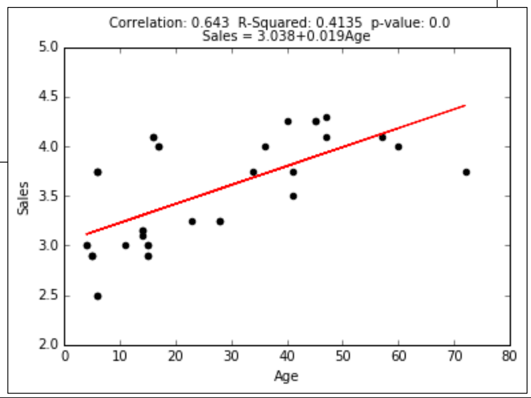
```
slope, intercept, r_value, p_val, std_err = stats.linregress(y=y,x=x)
print("This is regression with Ho: X does not help to predict Y/The
        slope is 0")
if np.sign(slope) < 1:
    slsign = ""
else:
    slsign = "+"
regeq = yvar + " = " + str(round(intercept,3)) + slsign +
         str(round(slope,3)) + xvar
print("The equation is " + regeq)
print("The R-Squared is " + str(round(r_value**2,4))+ " and the p-value
        is " + str(round(p_val,4)))
if p_val < alpha:
    print("Conclusion: Reject Ho: X does help predict Y")
else:
    print("Conclusion: Fail to Reject Ho: We can't reject that X
            doesn't help to predict Y")
```

```
This is regression with Ho: X does not help to predict Y/The slope is 0
The equation is Sales = 3.038+0.019Age
The R-Squared is 0.4135 and the p-value is 0.0
Conclusion: Reject Ho: X does help predict Y
```

Regression – Simple - Scatterplot

The scatterplot is a great way to examine relationships between two variables (Y and one X.)

```
plt.scatter(x,y,color='black')
xyCorr = round(x.corr(y),3)
plt.suptitle("Correlation: " + str(xyCorr)+ "  R-Squared: " +
        str(round(r_value**2,4))+ "  p-value: " + str(round(p_val,4)))
plt.title(regeq, size=10)
predict_y = intercept + slope * x
plt.plot(x,predict_y,'r-')
plt.xlabel(xvar)
plt.ylabel(yvar)
plt.savefig('RegressionOneX.png',
            bbox_inches='tight')
plt.show()
```



Final Conclusion:
There is a relationship between Age and Sales. Age explains 41% of the variability of Sales. As Age goes up by one year, Sales increase by .019 or 2 cents.

Regression - Simple

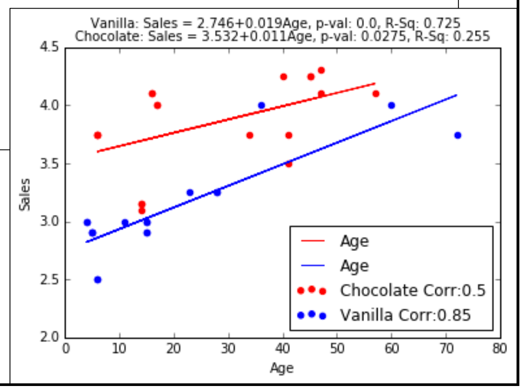
Suppose we want to see how Age predicts Sales, but also take into account Flavor. We can create two regressions.

```
xvar='Age'
yvar='Sales'
x1var='Chocolate'
x2var='Vanilla'
x1=ICData[xvar][ICData['Flavor']==x1var]
y1=ICData[yvar][ICData['Flavor']==x1var]
x2=ICData[xvar][ICData['Flavor']==x2var]
y2=ICData[yvar][ICData['Flavor']==x2var]

#Run the regression code for x1 and y1 and then also for x2 and y2 (in .py file)
```

```
Chocolate
This is regression with Ho: X does not help to predict Y/The slope is 0
The equation is Sales = 3.038+0.019Age
The R-Squared is 0.2549 and the p-value is 0.0275
Conclusion: Reject Ho: X does help predict Y

Vanilla
This is regression with Ho: X does not help to predict Y/The slope is 0
The equation is Sales = 3.038+0.019Age
The R-Squared is 0.7252 and the p-value is 0.0
Conclusion: Reject Ho: X does help predict Y
```

Regression – Simple - Scatterplot

```
OneCorr = round(np.corrcoef(x1,y1)[0,1],2)
TwoCorr = round(np.corrcoef(x2,y2)[0,1],2)
plt.scatter(x1,y1,color='red',label=x1var + ' Corr:'+str(OneCorr))
plt.scatter(x2,y2,color='blue',label=x2var + ' Corr:'+str(TwoCorr))
predict_y1 = intercept1 + slope1 * x1
predict_y2 = intercept2 + slope2 * x2
plt.plot(x1,predict_y1,'r-')
plt.plot(x2,predict_y2,'b-')
plt.title(x1var + ': ' + regeq1 + ', p-val: ' + str(round(p_val1,4)) +
         ', R-Sq: ' + str(round(r_value1**2,3)),size=10)
plt.suptitle(x2var + ': ' + regeq2 + ', p-val: ' + str(round(p_val2,4))+
         ', R-Sq: ' + str(round(r_value2**2,3)),size=10)
plt.xlabel(xvar)
plt.ylabel(yvar)
plt.legend(loc='best')
plt.savefig('RegressionwithCatX.png',
             bbox_inches='tight')
plt.show()
```



Final Conclusion:
Vanilla shows a strong relationship (correlation .85) between Age and Sales with 73% explained variance in Sales, but Chocolate has much weaker relationship (p-val .03).

If you have four sample means, which statistical procedure should you use to compare the means?

1. Analysis of Means
2. Analysis of Variance
3. Two-Sample t test
4. Scatterplot
5. Regression

DANIELS
COLLEGE OF BUSINESS
UNIVERSITY of DENVER

If you want to see if Sales is predicted by Customer Age, which statistical procedure should you use?

1. Analysis of Means
2. Analysis of Variance
3. Two-Sample t test
4. Cluster Analysis
5. Regression

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

What is the conclusion for the two sample t test?

1. Means for Children and Teens are the same
2. Means for Children and Teens are different

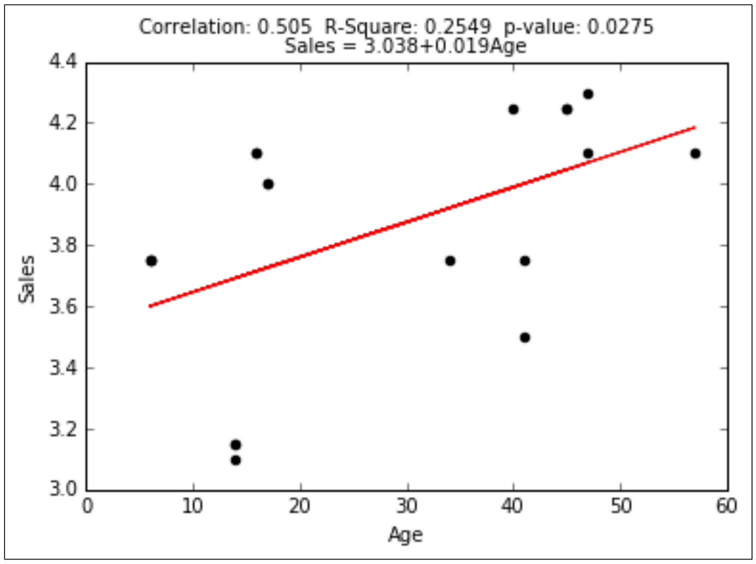
```
tmean, p_valmean =
     stats.ttest_ind(Child['Sales'],Teen['Sales'],
     equal_var=tEqVar)
```

```
This is a Two-Sample t test (assuming equal variances) of
equal means with Ho: The group means are equal
The t test statistic is -1.268 and the p-value is 0.2194
```

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER

In the following regression, what would the Type I Error (alpha) have to be for us to conclude the slope is 0?

1. .50
2. .30
3. .05
4. .01



REWIND and REV UP (optional)

REWIND

- Additional Practice Problems + Extra Credit

REV UP

- Similarities between t test for 2 means, ANOVA, and regression

DANIELS COLLEGE OF BUSINESS UNIVERSITY of DENVER