Science of Decisions
Interactive Stroop Effect Experiment

Analysis Results for Udacity P1 Study

Draft for Review- Revision 0.2

Prepared for: General Publication
Prepared by:

Laurie S. Reynolds
San Jose, California
laurie_reynolds@yahoo.com

Reviewed by: Udacity P1 Evaluator

Table of Contents

Revisions

Date	Revision	Notes
27 November, 2015	0.1	Initial submission for review
	0.2	Addressed reviewers comments Introduction ● Set of Hypothesis - Clarified that it applies to the population means Methods ● Figure 1 - Replaced line plot with bar chart ● Figure 2 - Added markers for the means Results ● Removed Figures 3 and 4 ● Updated Figure 5 to include the mean difference ● Corrected p-value in text and related figure ● Corrected Confidence Interval Discussion ● Added discussion of quantifiable test to determine normality and non-parametric test for further analysis ● Expanded discussion to include additional references and cases where the Stroop effect was studies. Appendix F ● Replaced with updated snapshot of IPython notebook

Abstract

In 1935, John Ridley Stroop published the results of an experimental study on the interference in verbal reactions [Ref 1] . Stroop found that conflicting word stimuli had a considerable interference effect. In the study, subjects are asked to say out loud the name of a color that is displayed under two different conditions. The time to say out loud a series of names was measured in seconds. In one condition, the color is displayed in the color name., (e.g the word Red is displayed in red.) In the other condition, the color is displayed in a name of a different color. (e.g. the word Green is displayed in red.)

In this study, the Stroop experiment was recreated using an online interactive version. Timing data from participants were analyzed using a repeated measures right tailed t-test. The analysis results corroborated Stroop's original findings that the interference due to word stimuli is significant.

Introduction

Dr. Eric Chudler, University of Washington, has made available an online interactive version of the Stroop experiment. [Ref. 3] For the purposes of this study, we were provided the timing data for 24 participants under each condition. Appendix A provides background for the study. Appendix B outlines the study protocol.

This investigation aims to determine if the word interference is significant in Chudler's interactive version of the experiment.

QFI #1: Independent and Dependent Variables

What is the independent variable?
The independent variable is the display condition (Congruent/Incongruent) for the colors. The congruent set contains colors that are displayed with the matching name. The incongruent set contains colors that are displayed with the text of colors that do not match. **Independent Variable**: Display condition: Congruent/Incongruent display of colors with color names

What is the dependent variable?
The dependent variable is the reaction time for a participant to read all the colors presented. This is measured in seconds. **Dependent Variable**: Reaction time in seconds

QFI #2: Set of Hypotheses

What is the appropriate set of hypotheses for this task?
In the original study, the Incongruent condition was found to be significantly more than the Congruent condition. This will form the basis for our hypotheses. Through analysis of the sample data, we will determine if we reject or accept the null hypothesis for the population. If the results follow the original study, the null hypothesis will be rejected in favor of the alternative hypothesis.

Null Hypothesis	Alternative Hypothesis	Number of Tails
$H_0 : \mu_{incongruent} <= \mu_{congruent}$ The population mean time for the Incongruent condition is less than or equal to the population mean time for the Congruent condition for the Stroop experiment.	$H_1 : \mu_{incongruent} > \mu_{congruent}$ The population mean time for the Incongruent condition is greater than the population mean time for the Congruent condition for the Stroop experiment.	1 Single, right tail

What kind of statistical test do you expect to perform?
In this study, timing data is from 24 participants that took the test under the two conditions. We do not have any population statistics. Given these conditions, we will use a repeated measure right tailed t-test **Statistical Test**: Repeated measures right tailed t-test

Methods

QFI #3: Descriptive Statistics

Report some descriptive statistics regarding this dataset. Include at least one measure of central tendency and at least one measure of variability

Table 1 shows measures of central tendency and variability for the data along with the calculated difference and percent difference.

As you can see the means and medians for each set are fairly close to each other which indicate that the data is relatively normally distributed. Additionally, for all the data except the Incongruent, the min and max are each within one to two standard deviations from the mean.

As we are estimating the population standard deviation, we apply Bessel's correction when calculating the sample standard deviation

In addition, we examine what percentage of the original time the difference represents in the Percent Diff column. We see that the average percentage difference is 63.42%. In other words, it takes an average of 63.42% longer to say out loud the names of colors in the Incongruent condition than under the Congruent condition.

[Ref 4 is a link to the Github repository which contains the IPython notebook that was used to perform the analysis. It also contains an excel spreadsheet which was used to double check the calculations.

Appendix F contains a copy of the IPython notebook used to perform analysis

Table 1: Descriptive Statistics

	Congruent	Incongruent	Difference	Percent Diff
count	24	24	24	24
mean	14.05	22.02	7.96	63.42
median	14.36	21.02	7.67	63.42
min	8.63	15.69	1.95	9.83
max	22.33	35.26	21.92	177.21
sstd	3.56	4.80	4.86	42.32
se	0.73	0.98	0.99	8.64

QFI #4: Supporting Visualizations

Provide one or two visualizations that show the distribution of the sample data. Write one or two sentences noting what you observe about the plot or plots
Figures 1 represents the raw data which is listed in table form in Appendix D. The data is sorted by Congruent timings. As we can see, the Incongruent timings are all greater than the congruent timings. The differences between the Congruent and corresponding Incongruent times vary in the size of the difference with two distinct outliers. Figure 2 shows the Congruent and Incongruent conditions overlaid on the same graph. This shows the differences between the means as well as the difference between the spread of the distribution of the values. Figure 3 shows the Differences histogram with probability density function overlay. The Incongruent and Differences show outliers, but for the purposed of this study, we will treat the data as normally distributed. See the discussion section on further analysis.

Figure 1: Raw data ordered by Congruent condition

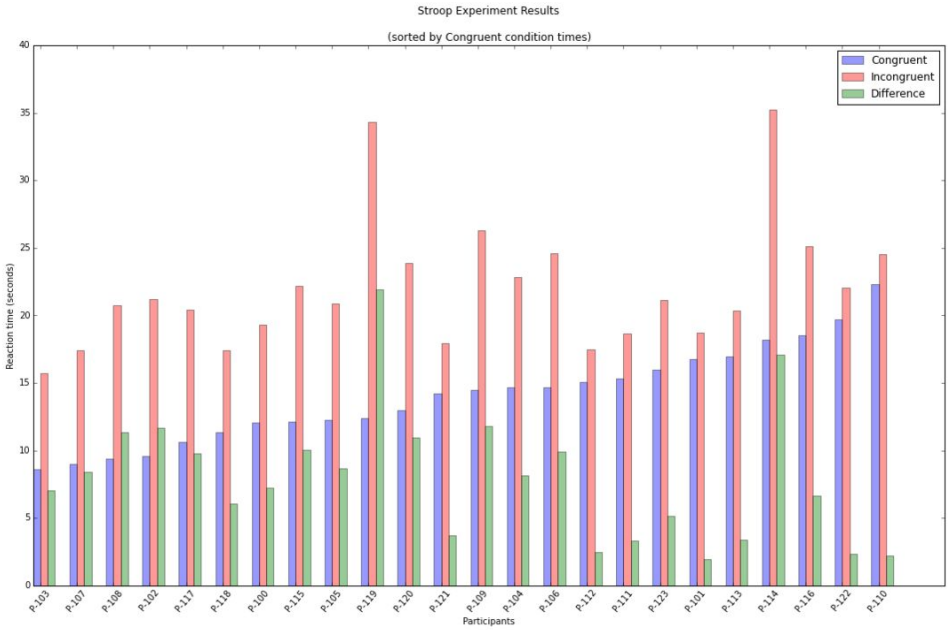


Figure 2: Congruent and Incongruent histograms

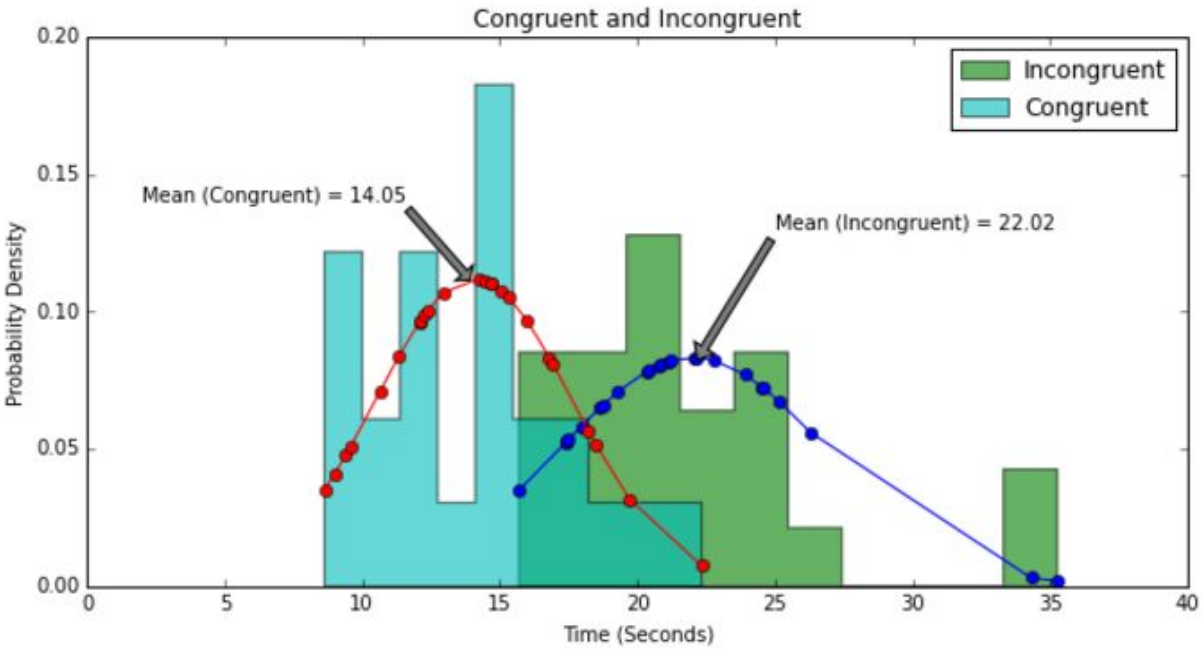
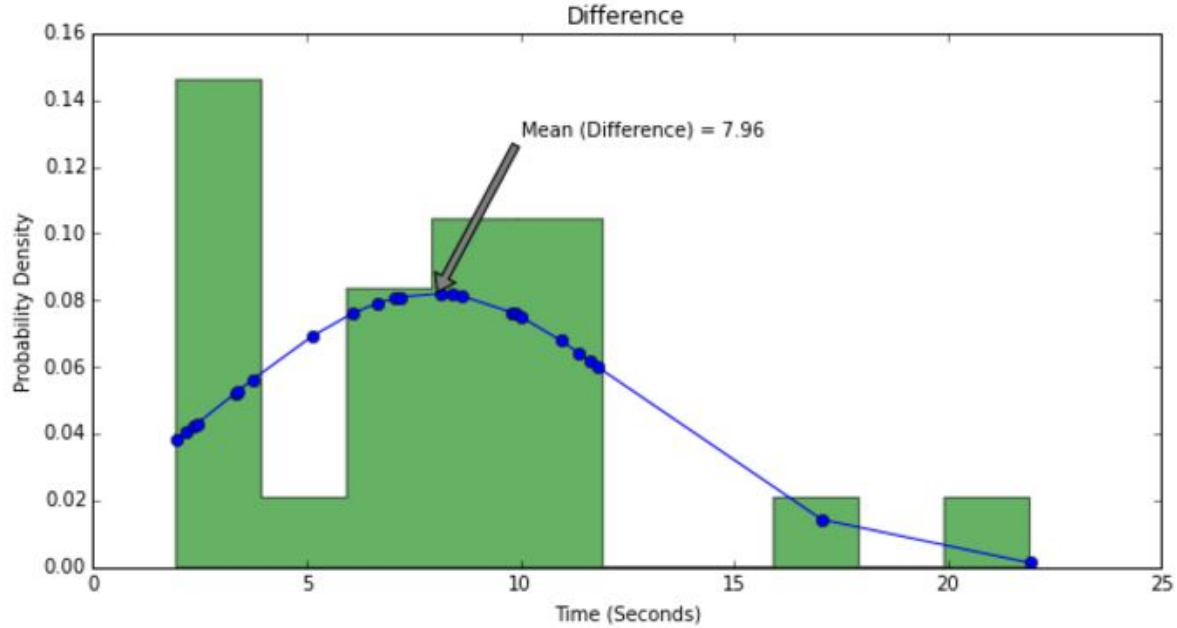


Figure 3: Timing Differences histogram

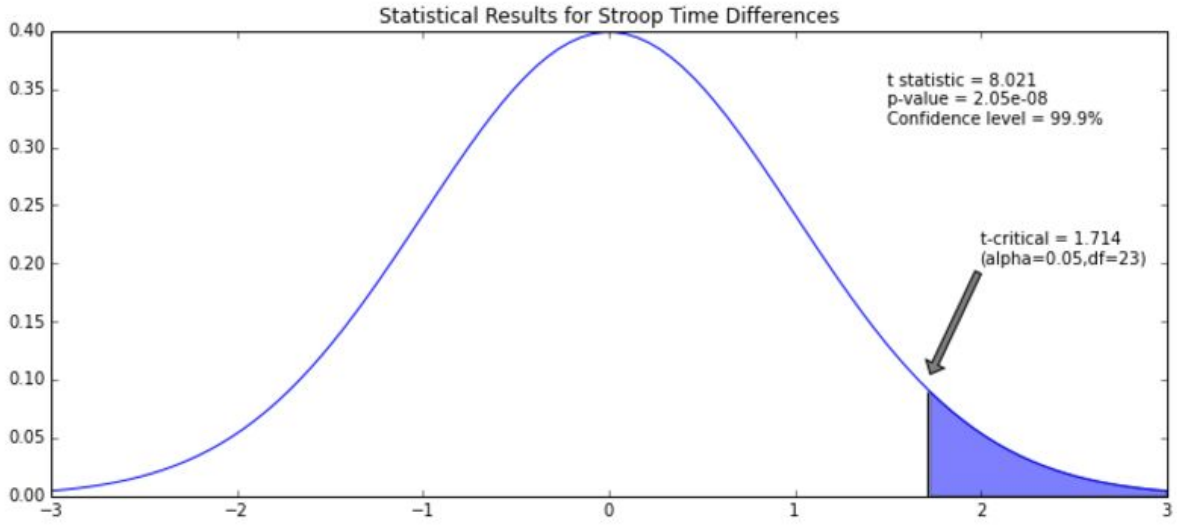


Results

QFI #5: Report Results

Now. perform the statistical test and report your results
Figure 6 shows the t critical region for $\alpha = 0.05,\ df = 23$. This illustrates that the t -statistic of 8.02 is much greater than the t-critical value of 1.71. t(23) = 8.02, p-value = 2.05e-8, right-tailed Confidence interval on the mean difference; 99.9%, CI = (6.26 to ∞) The effect size measures of Cohen's d = 1.64 and $r^2 = .74$ further demonstrate results are significant. Cohen's d = 1.64 r^2 = .74

Figure 4: t-critical region



What is your confidence level and your critical statistical value?
Confidence Level: 99.9% **Critical Statistic Value**: 1.714, $\alpha = 0.05,\ df = 23$

Do you reject the null hypotheses or fail to reject it?
Given the strong statistical effect size measures, we reject the null hypothesis that $\mu_{incongruent} <= \mu_{congruent}$ and accept our alternative hypothesis that $\mu_{incongruent} > \mu_{congruent}$

Come to a conclusion in terms of the experiment or task.
Word stimuli has a significant effect on the timing of saying out loud color names.

Did the results match up with your expectations
This has corroborated Stroop's findings from the original study and matches our expectations.

Discussion

QFI # 6: Digging Deeper

What do you think is responsible for the effects observed?
It appears that our verbal reading abilities are more dominant and faster than the recall abilities for the names of colors. Hoi-Chung Leung, et al in "An Event-related Functional MRI Study of the Stroop Color Word Interference Task" [Ref 2] measured the brain activity while a participant was performing the Stroop experiment. They found different areas of the cortex were activated when performing the test under different conditions.

Can you think of an alternatives or similar task that would result in a similar effect?
For an alternative to be effective, it would need to be universally recognized and the choices should be similar enough so that there is room for interference if labeled differently. The following are alternatives to consider ● Fruits (apple, oranges, pears, grape, melon) ● Emotions (happy, mad, sad, none)

The following might not be good candidates as the association with the shape may be stronger than the name.

- Common signs (stop, man, woman, yield)
- Transportation (car, plane, boat, bus, truck)

There have been some studies investigating the Stroop effect with other senses, specifically taste and auditory. In "Auditory Conflict Processing - Behavioral and Electrophysiologic Manifestations of the Stroop Effect", Henkin, et al. [Ref 6.] explore auditory interference through adjusting the speaker's gender and associated gender words by measuring participants reaction time to recognition of spoken words. They found that the Stroop effect did apply to auditory behavior.

Matching meaning and verbal tone is important part of communication. Alerting someone of an imminent danger is best conveyed with a loud, fast cadence. Another possible test could be to vary the sound and cadence of words and measure the reaction time to registering the meaning of the words.

White and Prescott describe a different stroop effect with taste identification in "Chemosensory Cross-Modal Stroop Effects: Congruent Odors Facilitate Taste Identification" [Ref 5]. They found that exposing participants to grapefruit led to faster identification of tastes associated with citric acid than exposing participants to a sweet smell, such as strawberries. The participants were asked to smell either a strawberry or grapefruit prior to tasting and identifying the food item. This study was different in that the congruent or incongruent condition was experienced before the test rather than simultaneously with the food item being evaluated.

Much of our lives rely on a synchronization of elements whether in written or verbal communication. In "Elements of Navigation", Silfver makes a connection between the Stroop effect and use of icons along with words in navigation. Pointing out picture/word interference impact to the usability of navigation. The Stroop test could be viewed as an early usability test.

Observations on the study design

The following factors of the study design may have an impact on the study results.

1. Self reporting of results. The study relies on participants saving and recording the timing results. Potential errors that could occur are as follows:
    - Incomplete reporting. This is seen by the difference in the number or results for each set.

- Swapping of values for the incongruent and congruent cases.
- Estimation of values. In cases where the participant did not remember to record the value presented, some may enter an estimate for completeness.
- Adjustment of values. There is the potential that a participant could adjust the values for what they feel may be expected for the study.

2. Manual calculation of differences: The study relies on participants calculating the difference between the two timings. Potential errors that could occur are as follows:
   - Miscalculation. There are two types of miscalculations that could occur.
     i. Incorrect difference value
     ii. Swapping the operands during the subtraction.
3. Entry of values. The recording of timings is done by selecting from a set of choices that describe time ranges. It is possible that a participant could accidently select a choice that does not correspond to the actual values.
4. Multiple results in one session. The study does not detect whether or not a participant takes the test multiple times in one session which could impact the results

Further analysis recommendation

For the purposes of this study, we treated the data as normally distributed. This seems to be further supported by looking at the Quantile-Quantile (QQ) plots which compares the probability distributions for each condition to the normal curve. R^2 values greater than 80% seems to confirm this. However, some would argue that QQ plots should not be used on less than one thousand data points..

Figure 5: Probability Plot for Congruent condition

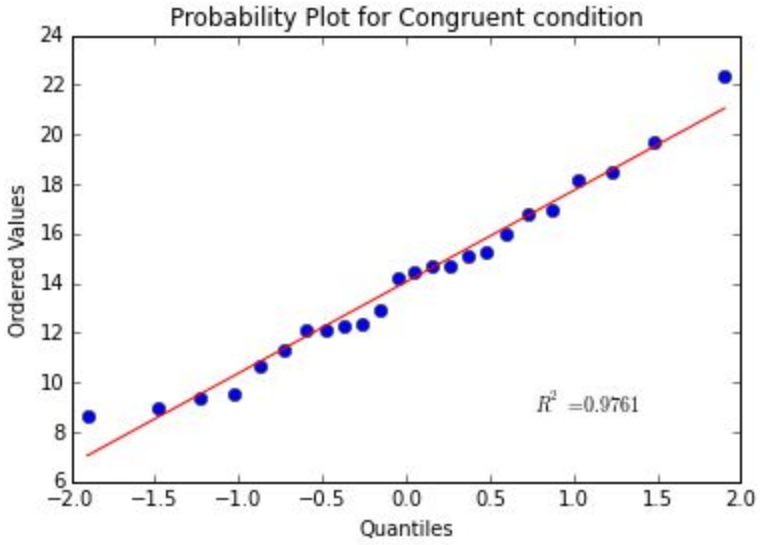


Figure 6: Probability Plot for Incongruent condition

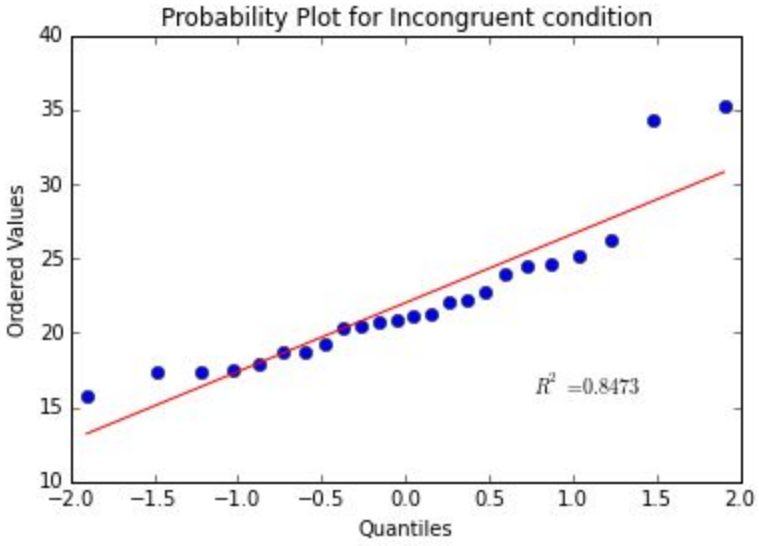
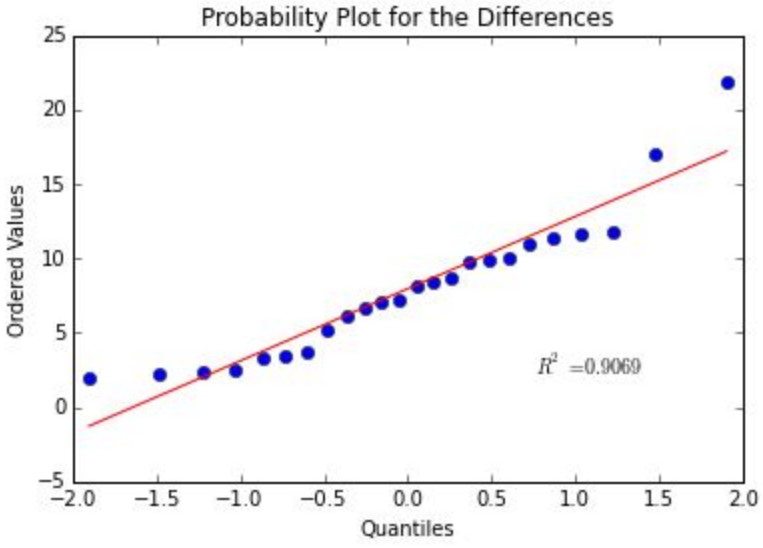


Figure 7: Probability Plot for Differences



As was pointed out by the reviewer, a more accurate and quantifiable test to determine normality is the Shapiro-Wilk test. Running this, we determine that the Congruent condition is normally distributed, but the Incongruent condition is not.

Table 2: Shapiro-Wilk results for the null hypothesis that the distribution is normal

Condition	Shapiro-Wilk value	p-value	Reject the null $\alpha = 0.05$
Congruent	0.97	0.69	Fail to reject the null p-value is > 0.05
Incongruent	0.85	0.003	Reject the null p-value < 0.05

As the Shapiro-Wilk test demonstrated that the data is not normally distributed for the Incongruent condition, a non-parametric test is recommended for further analysis. As the data is paired, the Wilcoxon Signed Ranks test is an option. As the sample set is less than 25, the small sample form of the Wilcoxon Signed Ranks test should be used.

References

1. Stroop, John Ridley (1935). “Studies of Interference in Serial Verbal Reactions”, *Journal of Experimental Psychology*, Vol 18(6), Dec 1935, 643-662. Retrieved from: York University Classics in the History of Psychology - http://psychclassics.yorku.ca/Stroop/

2. Hoi-Chung Leung, Pawel Skudlarski, James C. Gatenby, Bradley S. Peterson, and John C. Gore. “An Event-related Functional MRI Study of the Stroop Color Word Interference Task”. *Cerebral Cortex* (2000) 10 (6): 552-560.doi: 10.1093/cercor/10.6.552 Retrieved from Cerebral Corted - http://cercor.oxfordjournals.org/content/10/6/552.full

3. Chudler, Eric H. Executive Director, Center for Sensorimotor Neural Engineering Sources

    - Interactive Stroop Effect Experiment: https://faculty.washington.edu/chudler/java/ready.html
    - Stroop Effect: https://faculty.washington.edu/chudler/words.html#seffect

4. Github repository for analysis artifacts:

    https://github.com/golgistudio/udacity-data-analyst-stroop

5. White, Theresa L. and Prescott, John. “Chemosensory Cross-Modal Stroop Effects: Congruent Odors Facilitate Taste Identification”, Retrieved from: Chem. Senses (2007) 32 (4): 337-341.doi: 10.1093/chemse/bjm001 First published online: February 16, 2007

    http://chemse.oxfordjournals.org/content/32/4/337.full.pdf

6. Henkin Y, Yaar-Soffer Y, Gilat S, Muchnik C. (2010) Auditory Conflict Processing - Behavioral and Electrophysiologic Manifestations of the Stroop Effect. *Journal of the American Academy of Audiology* 21(7):474-486

    http://connection.ebscohost.com/c/articles/53988712/auditory-conflict-processing-behavioral-electrophysiologic-manifestations-stroop-effect

    http://www.audiology.org/news/applying-stroop-effect-audition

7. Silfver, Peter. “Elements of Navigation”, Smashing Magazine http://www.smashingmagazine.com/2012/03/the-elements-of-navigation/

8. Wilcoxon Signed Ranks test - statistic lectures http://www.statisticslectures.com/topics/wilcoxonsignedranks/

Appendix A: Study Instructions and Rubric

These instructions were provided by Udacity

Background Information
In a Stroop task, participants are presented with a list of words, with each word displayed in a color of ink. The participant's task is to say out loud the *color of the ink* in which the word is printed. The task has two conditions: a congruent words condition, and an incongruent words condition. In the *congruent words* condition, the words being displayed are color words whose names match the colors in which they are printed: for example RED, BLUE. In the *incongruent words* condition, the words displayed are color words whose names do not match the colors in which they are printed: for example PURPLE, ORANGE. In each case, we measure the time it takes to name the ink colors in equally-sized lists. Each participant will go through and record a time from each condition.

Table: Study Rubric provided by Udacity

Study Rubric	
Criteria	**Meets Specifications**
Question 1: Identify variables in the experiment	Question response correctly identifies the independent and dependent variables in the experiment.
Question 2: Establish a hypothesis and statistical test	An appropriate hypothesis test has been stated along with an appropriate statistical test to apply to collected data, with appropriate justification.
Question 3: Report descriptive statistics	Descriptive statistics, including at least one measure of centrality and one measure of variability, have been computed for the dataset's groups.
Question 4: Plot the data	One or two visualizations have been created that show off the data, including comments on what can be observed in the plot or plots.
Question 5: Perform the statistical test and interpret your results	A statistical test has been correctly performed and reported, including test statistic, critical test statistic or p-value, and test result. The test results are interpreted in terms of the experimental task performed.
Question 6: Digging deeper and extending the investigation	Hypotheses regarding the reasons for the effect observed are presented. An extension or related experiment to the performed Stroop task is provided, that may produce similar effects. ***This question is optional and does not need to be answered in order to meet project specifications***

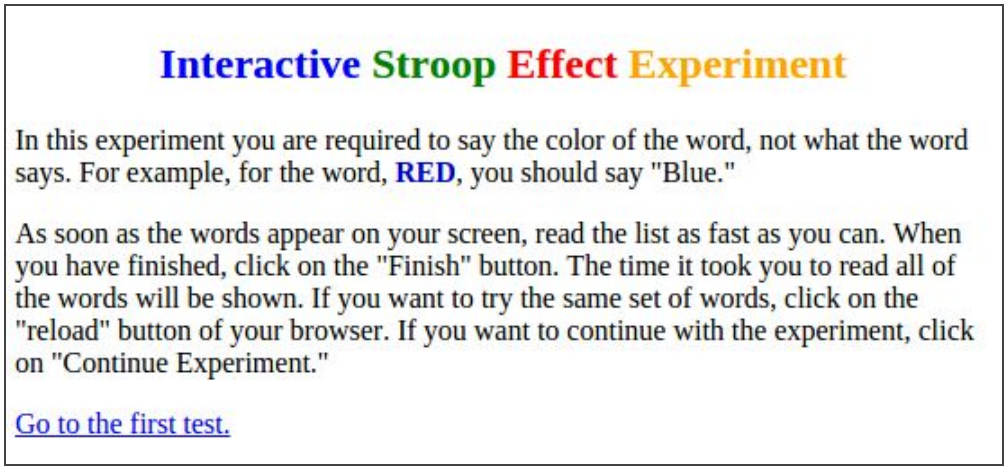
Appendix B: Experiment Protocol

The following outlines the Stroop Effect protocol used for this experiment.

1. Word set #1 is displayed - Congruent - 25 color names shaded with the same color as the name.
2. Participant reads each color out loud and selects [Finish] when complete.
3. The application displays the amount of time the page with word set #1 has been displayed.
4. Participant records the time.
5. Participant selects [Continue Experiment] to continue
6. Word set # 2 is displayed - Incongruent - The same color names are displayed. This time, they are shaded a color that does not match their names.
7. Participant says each color (not the text name) out loud and selects [Finish] when complete.
8. The application displays the amount of time the page with word set #2 has been displayed.
9. Participant records the time.
10. Participant selects [Enter and compare your results] to continue
11. Participant is asked to select the range in which each time falls and select [Submit] after each time is entered.
    - Word set #1
    - Word set #2
    - Difference - Participant is asked to calculate the difference by subtracting the first word set time from the second word set time.
12. After each time is submitted, the results for the current set of participants is displayed in tabular and pie chart form.

Screenshot from example run

1) Instructions: Participants are directed to navigate to the following URL to start the test: https://faculty.washington.edu/chudler/java/ready.html



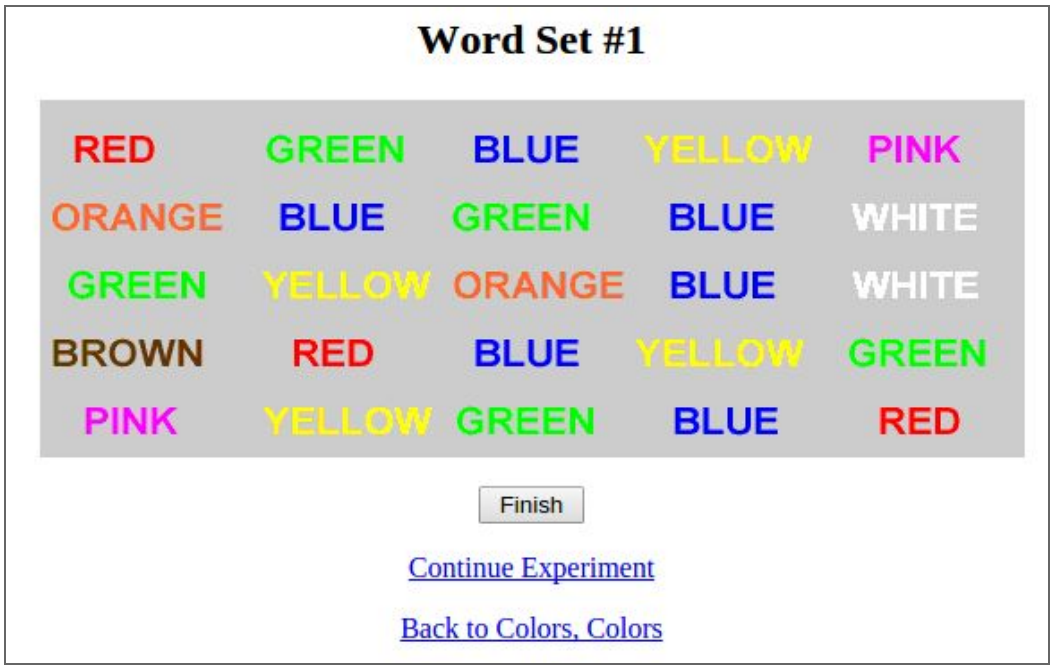
Interactive Stroop Effect Experiment

In this experiment you are required to say the color of the word, not what the word says. For example, for the word, **RED**, you should say "Blue."

As soon as the words appear on your screen, read the list as fast as you can. When you have finished, click on the "Finish" button. The time it took you to read all of the words will be shown. If you want to try the same set of words, click on the "reload" button of your browser. If you want to continue with the experiment, click on "Continue Experiment."

Go to the first test.

2) Word Set # 1 - https://faculty.washington.edu/chudler/java/timesc.html



Word Set #1

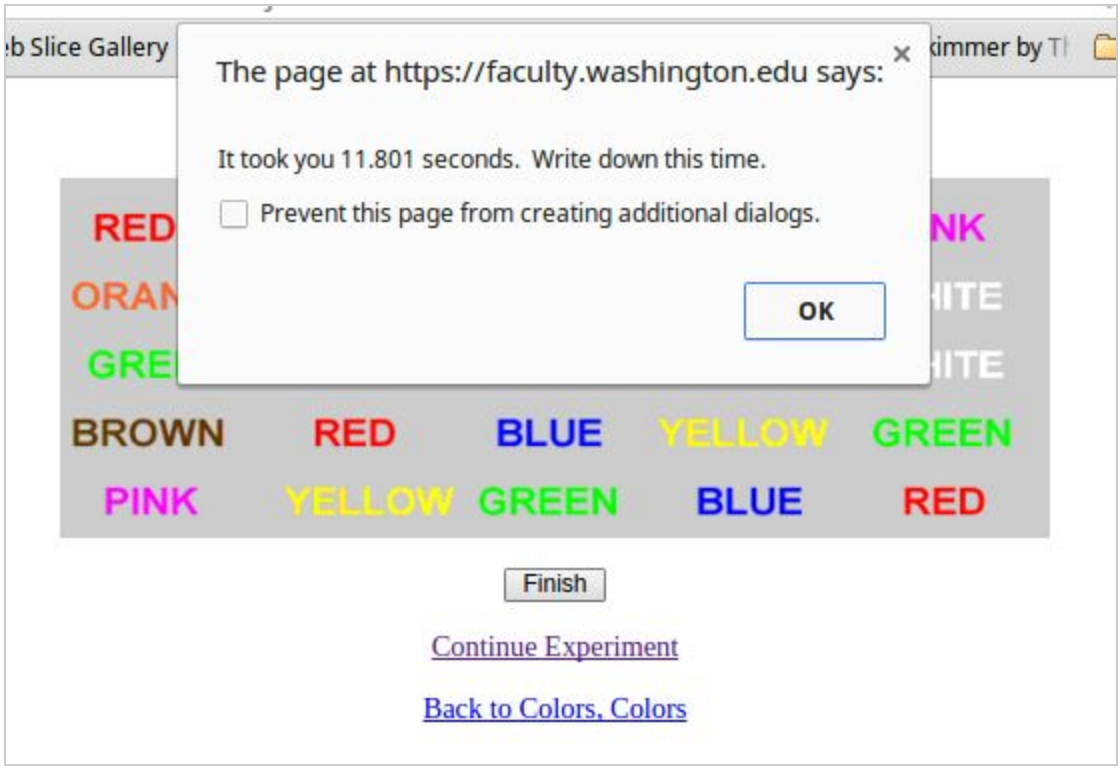
RED	GREEN	BLUE	YELLOW	PINK
ORANGE	BLUE	GREEN	BLUE	WHITE
GREEN	YELLOW	ORANGE	BLUE	WHITE
BROWN	RED	BLUE	YELLOW	GREEN
PINK	YELLOW	GREEN	BLUE	RED

Finish

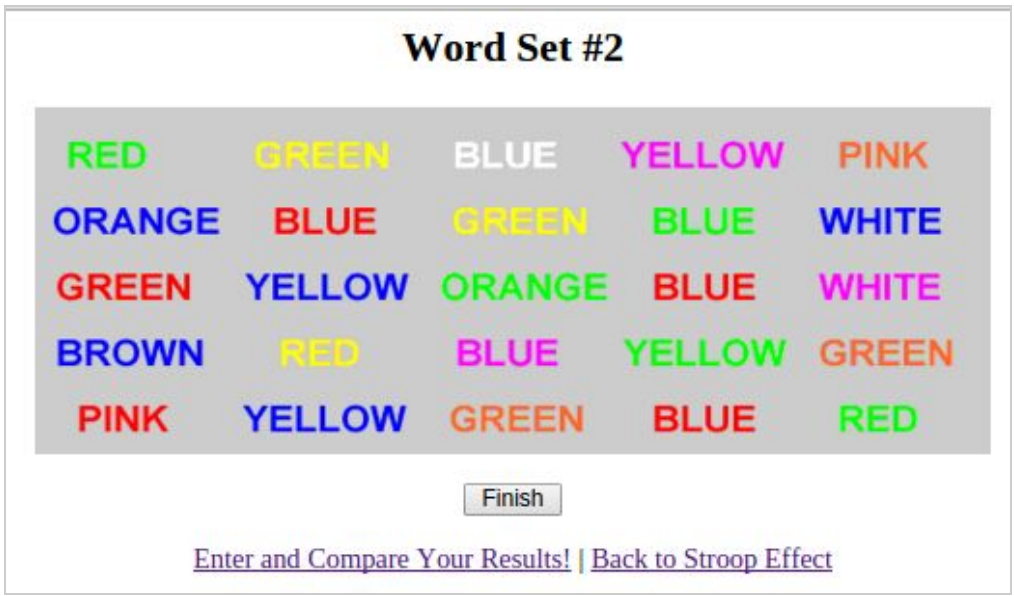
Continue Experiment

Back to Colors, Colors

3) Timing results for Word Set #1.

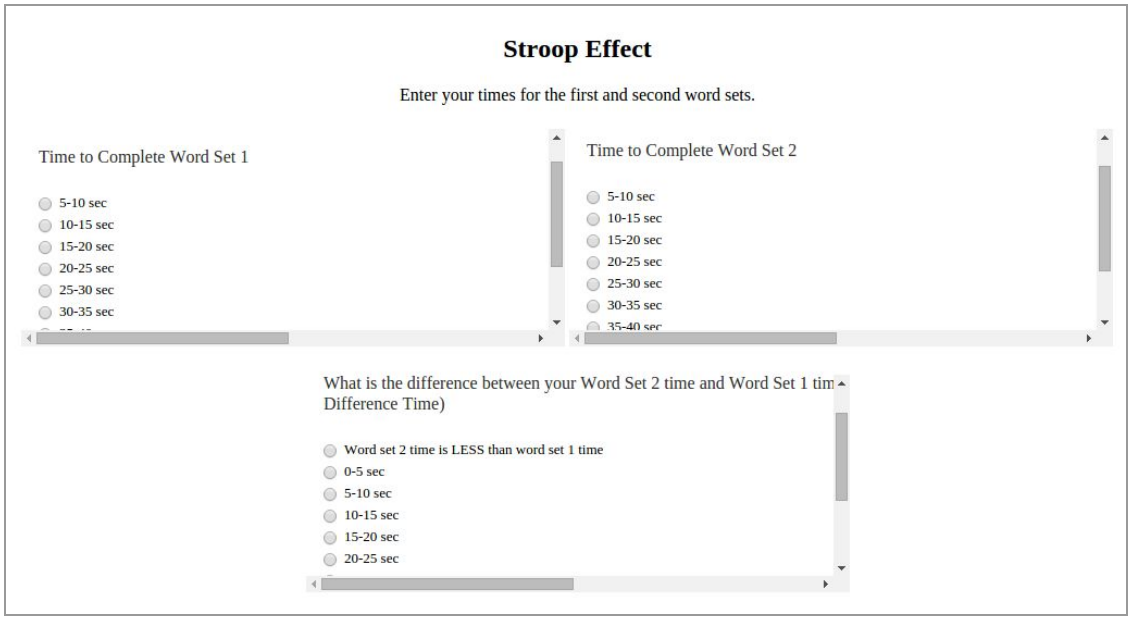


4) Word Set #2 - https://faculty.washington.edu/chudler/java/timestc.html



5) Timing results for Word Set #2

6) Record Results - https://faculty.washington.edu/chudler/java/strvote.html



7) After selecting [Submit] for each time entered, the cumulative results are displayed

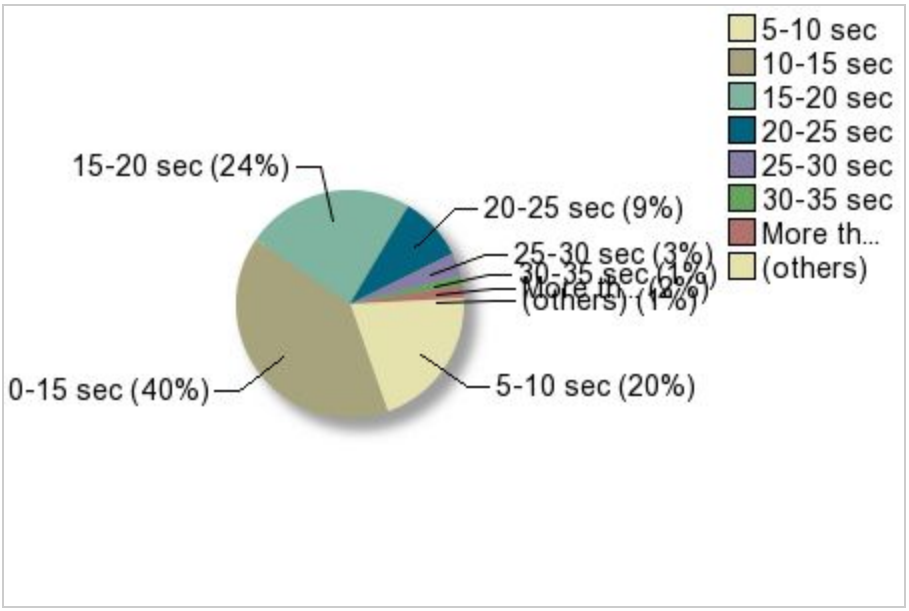
Appendix C: Poll results for all samples to date

(Recorded November, 2015 - 2:30 pm PST)

Time to Complete Word Set 1

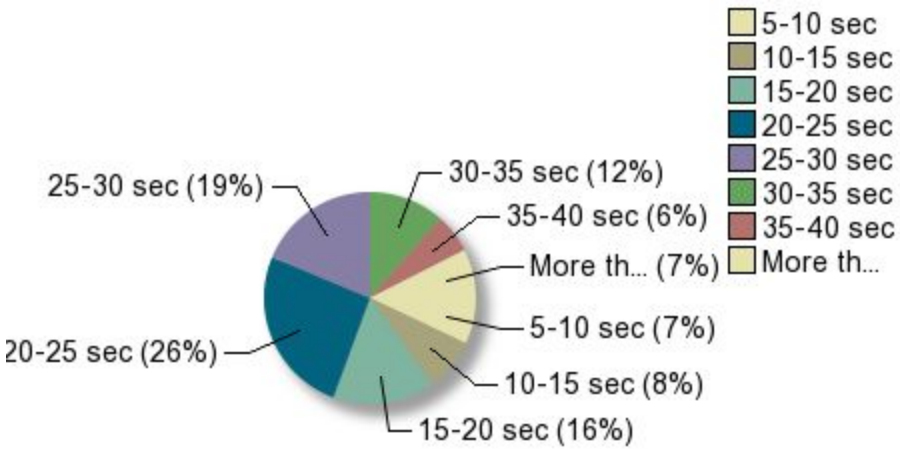
Poll Results
Total submissions: 105300

Answer choice	*Frequency*	*Percentage*
5-10 sec	20651	19.61%
10-15 sec	41991	39.88%
15-20 sec	25363	24.09%
20-25 sec	9640	9.15%
25-30 sec	3461	3.29%
30-35 sec	1457	1.38%
35-40 sec	849	0.81%
More than 40 sec	1888	1.79%



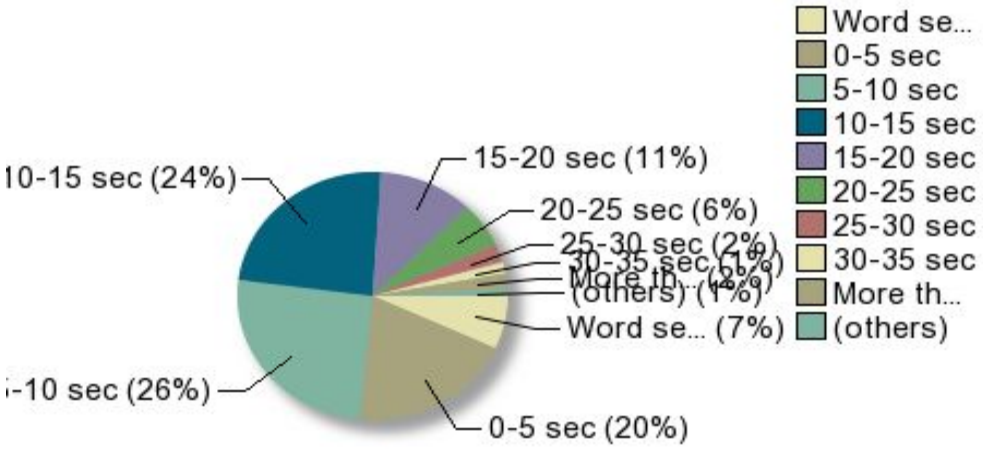
Time to Complete Word Set 2

Poll Results Total submissions: 105610		
Answer choice	Frequency	Percentage
5-10 sec	7442	7.05%
10-15 sec	8432	7.98%
15-20 sec	16448	15.57%
20-25 sec	27117	25.68%
25-30 sec	19771	18.72%
30-35 sec	12207	11.56%
35-40 sec	6303	5.97%
More than 40 sec	7890	7.47%



Poll Results for the differences

Poll Results Total submissions: 181097		
Answer choice	Frequency	Percentage
Word set 2 time is LESS than word set 1 time	12571	6.94%
0-5 sec	35421	19.56%
5-10 sec	46261	25.54%
10-15 sec	43427	23.98%
15-20 sec	20550	11.35%
20-25 sec	10617	5.86%
25-30 sec	4329	2.39%
30-35 sec	2683	1.48%
35-40 sec	1690	0.93%
More than 40 sec	3548	1.96%



Appendix D: Stroop Effect - Raw Data

Sample Results were provided by Udacity

Congruent	Incongruent
12.079	19.278
16.791	18.741
9.564	21.214
8.63	15.687
14.669	22.803
12.238	20.878
14.692	24.572
8.987	17.394
9.401	20.762
14.48	26.282
22.328	24.524
15.298	18.644
15.073	17.51
16.929	20.33
18.2	35.255
12.13	22.158
18.495	25.139
10.639	20.429
11.344	17.425
12.369	34.288
12.944	23.894
14.233	17.96
19.71	22.058
16.004	21.157

Appendix E: Data Preparation and Analysis Workflow

Stroop Sample Result Data
The sample result data of 23 samples - two columns (Congruent/Incongruent) was provided in the form of a comma separated values. The following steps were taken to prepare the data for analysis.

1. Assigned sequential participant ids starting with P-100
2. Calculated the difference between the Congruent and Incongruent timings using the built in function tools in Google Sheets
3. Saved data as a comma separated file
4. Converted data to JSON format using Convert CSV to JSON (see table Tools for tools used)
5. Analysis and visualizations done in Jupyter, a python development environemnt.
6. Duplicate calculations were performed in Google Sheets.

Table: Data Preparation and Analysis Tools

Data Preparation and Analysis Tools		
Tool name	**Source**	**Description**
CSV to JSON	http://www.convertcsv.com/csv-to-json.htm	
Google Sheets	https://www.google.com/sheets/about/	
Jupyter	http://jupyter.org/	Python development environment

Appendix F: Copy of IPython Notebook

In [1]:

```
#          Science of Decisions
#     Interactive Stroop Effect Experiment

#     Analysis Results for Udacity P1 Study

# Prepared by:  Laurie S. Reynolds
```

In [2]:

```
# Read in the data and display as a table


# Some initial imports and settings
%matplotlib inline

from IPython.display import display, HTML
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.mlab as mlab
import math as math
import scipy.stats as stats

np.set_printoptions(suppress=True)

# Read in the data and display as a table
stroopData = pd.read_csv('/home/laurie/udacity/P1/stroopdata-differences.csv')

stroopData
```

Out[2]:

	Participant	Congruent	Incongruent	Difference
0	P-100	12.079	19.278	7.199
1	P-101	16.791	18.741	1.950

2	P-102	9.564	21.214	11.650
3	P-103	8.630	15.687	7.057
4	P-104	14.669	22.803	8.134
5	P-105	12.238	20.878	8.640
6	P-106	14.692	24.572	9.880
7	P-107	8.987	17.394	8.407
8	P-108	9.401	20.762	11.361
9	P-109	14.480	26.282	11.802
10	P-110	22.328	24.524	2.196
11	P-111	15.298	18.644	3.346
12	P-112	15.073	17.510	2.437
13	P-113	16.929	20.330	3.401
14	P-114	18.200	35.255	17.055
15	P-115	12.130	22.158	10.028
16	P-116	18.495	25.139	6.644
17	P-117	10.639	20.429	9.790
18	P-118	11.344	17.425	6.081
19	P-119	12.369	34.288	21.919
20	P-120	12.944	23.894	10.950
21	P-121	14.233	17.960	3.727
22	P-122	19.710	22.058	2.348
23	P-123	16.004	21.157	5.153

In [7]:

```
# Data is not displaying correctly as numbers, verify check the data was imported as floats
stroopData.dtypes
```

Out[7]:

```
Participant     object
Congruent      float64
Incongruent    float64
Difference     float64
dtype: object
```

In [10]:

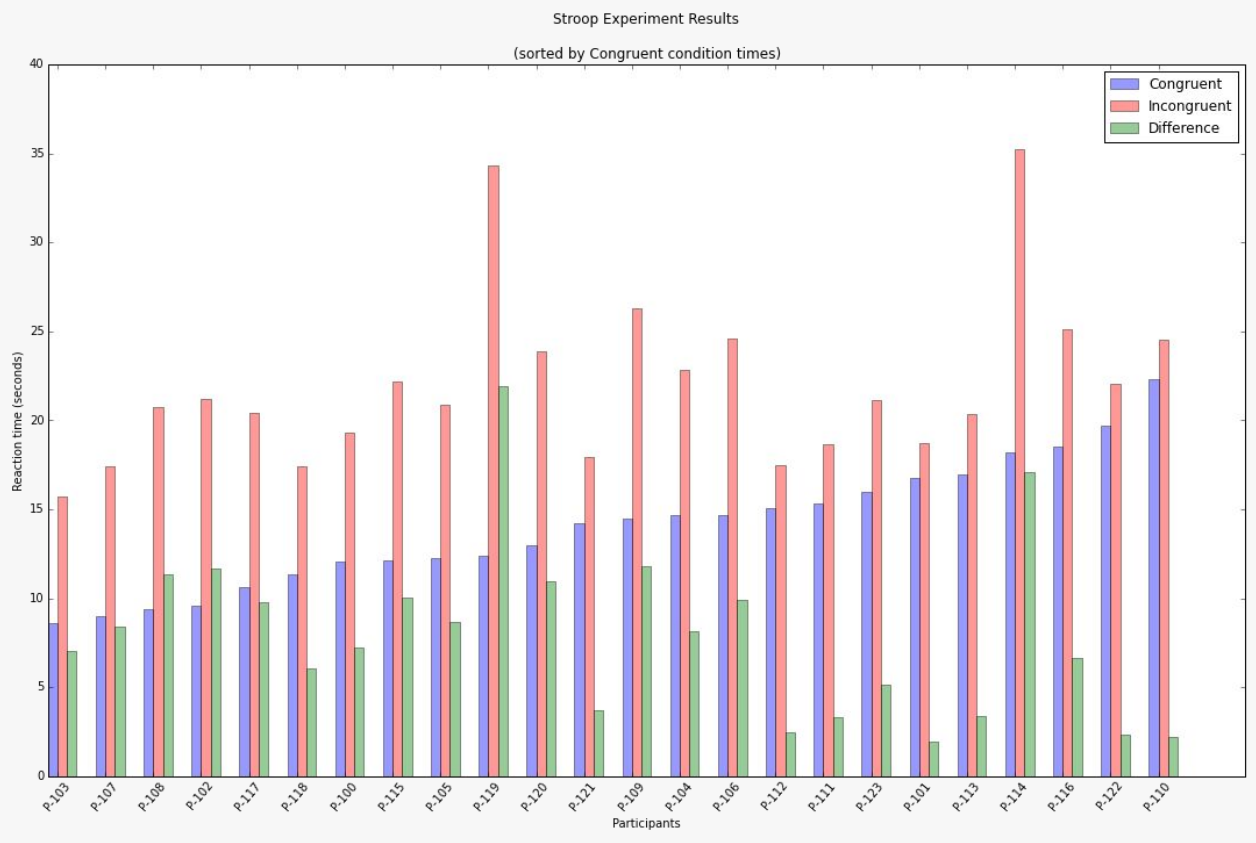
```
# Calculate the percentage difference

stroopData['Percent Diff'] = (stroopData['Difference'] / stroopData['Congruent'] ) * 100

# display the data formatted correctly
print (stroopData.to_string (justify='right', index=False))
```

```
Participant  Congruent  Incongruent  Difference  Percent Diff
      P-100     12.079       19.278       7.199     59.599305
      P-101     16.791       18.741       1.950     11.613364
      P-102      9.564       21.214      11.650    121.810958
      P-103      8.630       15.687       7.057     81.772885
      P-104     14.669       22.803       8.134     55.450269
      P-105     12.238       20.878       8.640     70.599771
      P-106     14.692       24.572       9.880     67.247482
      P-107      8.987       17.394       8.407     93.546233
      P-108      9.401       20.762      11.361    120.848846
      P-109     14.480       26.282      11.802     81.505525
      P-110     22.328       24.524       2.196      9.835185
      P-111     15.298       18.644       3.346     21.872140
      P-112     15.073       17.510       2.437     16.167982
      P-113     16.929       20.330       3.401     20.089787
      P-114     18.200       35.255      17.055     93.708791
      P-115     12.130       22.158      10.028     82.671063
      P-116     18.495       25.139       6.644     35.923222
      P-117     10.639       20.429       9.790     92.019927
      P-118     11.344       17.425       6.081     53.605430
      P-119     12.369       34.288      21.919    177.209152
      P-120     12.944       23.894      10.950     84.595179
      P-121     14.233       17.960       3.727     26.185625
      P-122     19.710       22.058       2.348     11.912735
      P-123     16.004       21.157       5.153     32.198200
```

In [13]:

```
# Sort the data by Congruent column
#stroopSorted = stroopData.sort_values(['Congruent', 'Incongruent', 'Difference', 'Participant',
```

```
'Percent Diff'],  ascending=[1, 0, 0, 0,0])
n_groups = 24

stroopSorted = stroopData.sort_values(by='Congruent')
stroopSorted.index = range(0,len(stroopSorted))

fig, ax = plt.subplots(1,1,figsize=(15,10))

index = np.arange(n_groups)
bar_width = 0.2

opacity = 0.4
error_config = {'ecolor': '0.3'}

rects1 = plt.bar(index, stroopSorted['Congruent'], bar_width,
                 alpha=opacity,
                 color='b',
                 label='Congruent')

rects2 = plt.bar(index + bar_width, stroopSorted['Incongruent'], bar_width,
                 alpha=opacity,
                 color='r',
                 label='Incongruent')

rects2 = plt.bar(index + 2*bar_width, stroopSorted['Difference'], bar_width,
                 alpha=opacity,
                 color='g',
                 label='Difference')


plt.xlabel('Participants')
plt.ylabel('Reaction time (seconds)')
plt.title('Stroop Experiment Results\n\n(sorted by Congruent condition times)')
plt.xticks(index + bar_width, stroopSorted['Participant'])
plt.xticks(rotation=50)
plt.legend()

plt.tight_layout()
plt.show()
```

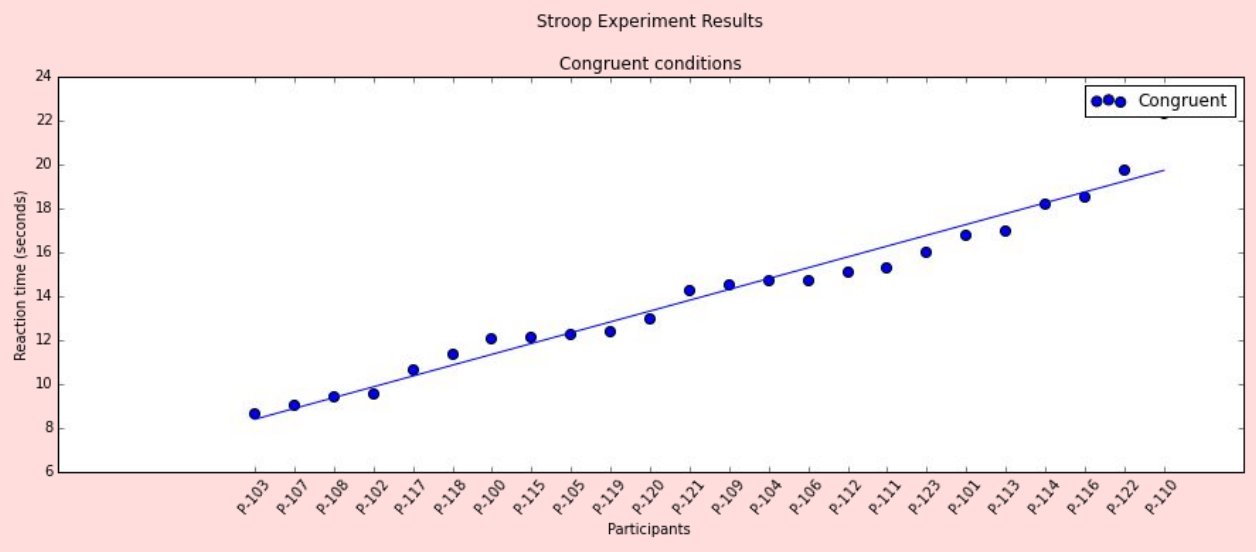


In [14]:

```
fig, ax = plt.subplots(1,1,figsize=(15,5))

plt.scatter(index, stroopSorted['Congruent'], s=50, c='blue', label='Congruent')

m, b = np.polyfit(index, stroopSorted['Congruent'], 1)

plt.plot(index, m*index + b, '-')


plt.title('Stroop Experiment Results\n\nCongruent conditions')

plt.xticks(index, stroopSorted['Participant'])
plt.xticks(rotation=50)

plt.xlabel('Participants')
plt.ylabel('Reaction time (seconds)')

plt.legend()
```

```
plt.show()
```

```
/usr/local/bin/anaconda2/lib/python2.7/site-packages/matplotlib/collections.py:590: FutureWarning:
elementwise comparison failed; returning scalar instead, but in the future will perform elementwise
comparison
  if self._edgecolors == str('face'):
```

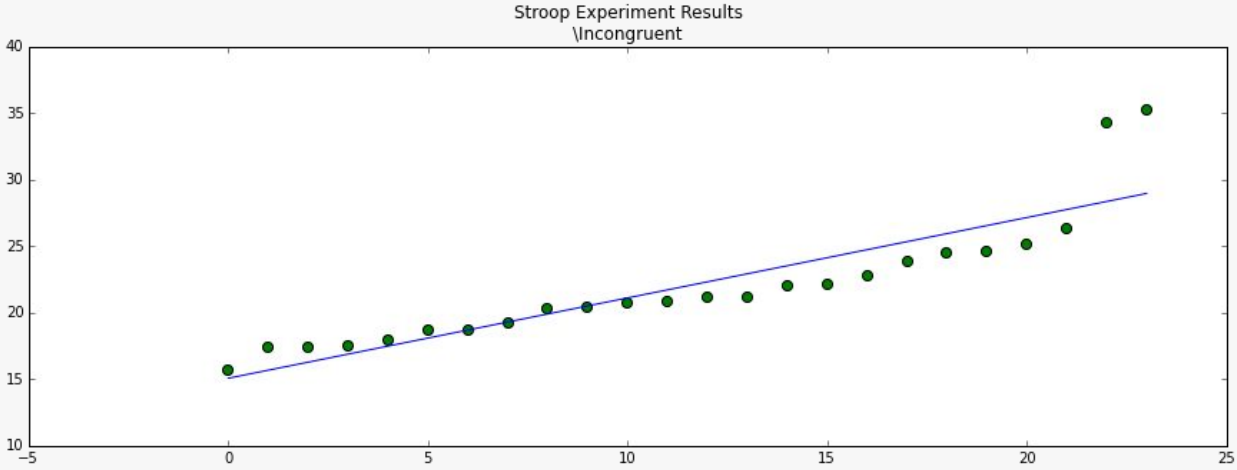


In [15]:

```
fig, ax = plt.subplots(1,1,figsize=(15,5))

stroopSortedIncongruent = stroopData.sort_values(['Incongruent', 'Congruent', 'Difference',
'Participant', 'Percent Diff'],  ascending=[1, 0, 0, 0,0])
n_groups = 24

ml, bl = np.polyfit(index, stroopSortedIncongruent['Incongruent'], 1)

plt.plot(index, ml*index + bl, '-')

plt.scatter(index, stroopSortedIncongruent['Incongruent'], s=50, c='green', label='Incongruent')

plt.title('Stroop Experiment Results\n\Incongruent')


plt.xticks(index, stroopSortedDifference['Participant'])
plt.xticks(rotation=50)

plt.xlabel('Participants')
plt.ylabel('Reaction time (seconds)')
```

```
plt.show()
```

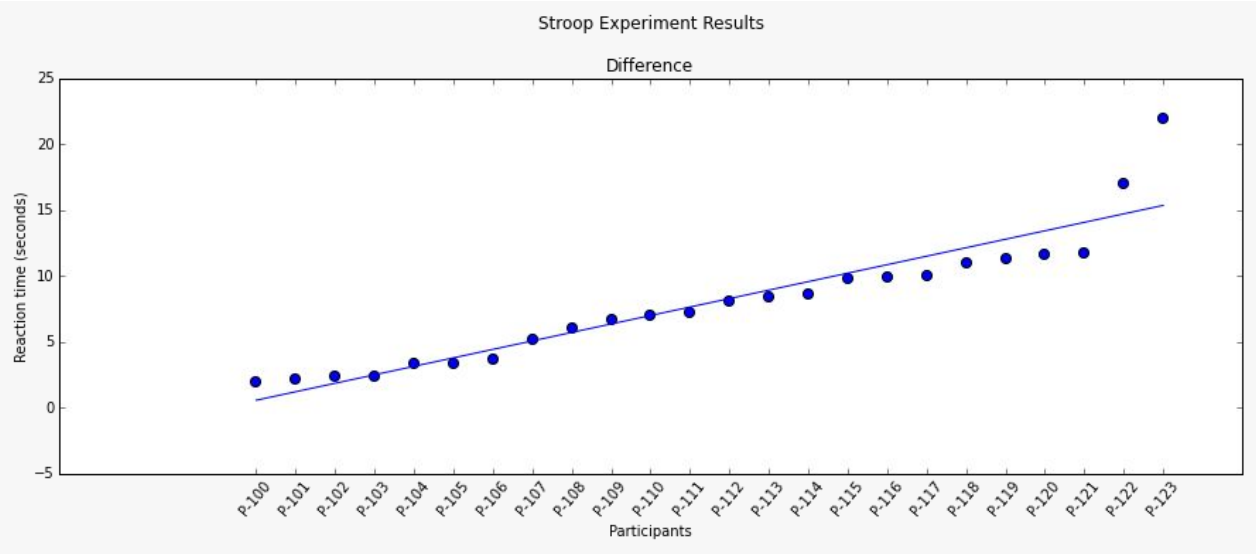
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-15-91aece9ff0c7> in <module>()
     13
     14
---> 15 plt.xticks(index, stroopSortedDifference['Participant'])
     16 plt.xticks(rotation=50)
     17

NameError: name 'stroopSortedDifference' is not defined
```



In [16]:

```
fig, ax = plt.subplots(1,1,figsize=(15,5))

stroopSortedDifference = stroopData.sort_values(['Difference', 'Congruent', 'Incongruent',
'Participant', 'Percent Diff'],  ascending=[1, 0, 0, 0,0])
n_groups = 24

m, b = np.polyfit(index, stroopSortedDifference['Difference'], 1)

plt.plot(index, m*index + b, '-')


plt.scatter(index, stroopSortedDifference['Difference'], s=50, c='blue')

plt.title('Stroop Experiment Results\n\nDifference')
```

```
plt.xticks(index, stroopSortedDifference['Participant'])
plt.xticks(rotation=50)

plt.xlabel('Participants')
plt.ylabel('Reaction time (seconds)')

plt.show()
```



In [17]:

```
# Cacluatle descriptive statistics

# Assign column data to variables
participantCol = stroopData['Participant']
incongruentCol = stroopData['Incongruent']
differenceCol = stroopData['Difference']
congruentCol = stroopData['Congruent']
percentDiffCol = stroopData['Percent Diff']

descriptiveStats = pd.DataFrame( columns=['Congruent','Incongruent','Difference', 'Percent Diff'],
index=['count', 'mean', 'median', 'min', 'max', 'std', 'se'])


# Calculate statistics
def calcStats(columnName, columnData) :

    descriptiveStats[columnName]['count'] = columnData.count()
    descriptiveStats[columnName]['mean'] = columnData.mean()
```

```
    descriptiveStats[columnName]['median'] = columnData.median()
    descriptiveStats[columnName]['min'] = columnData.min()
    descriptiveStats[columnName]['max'] = columnData.max()
    descriptiveStats[columnName]['std'] = stats.tstd(columnData)
    descriptiveStats[columnName]['se'] = stats.sem(columnData)


# Fill in the table
calcStats('Congruent', congruentCol)
calcStats('Incongruent', incongruentCol)
calcStats('Difference', differenceCol)
calcStats('Percent Diff', percentDiffCol)


# Print out the results
descriptiveStats
```

Out[17]:

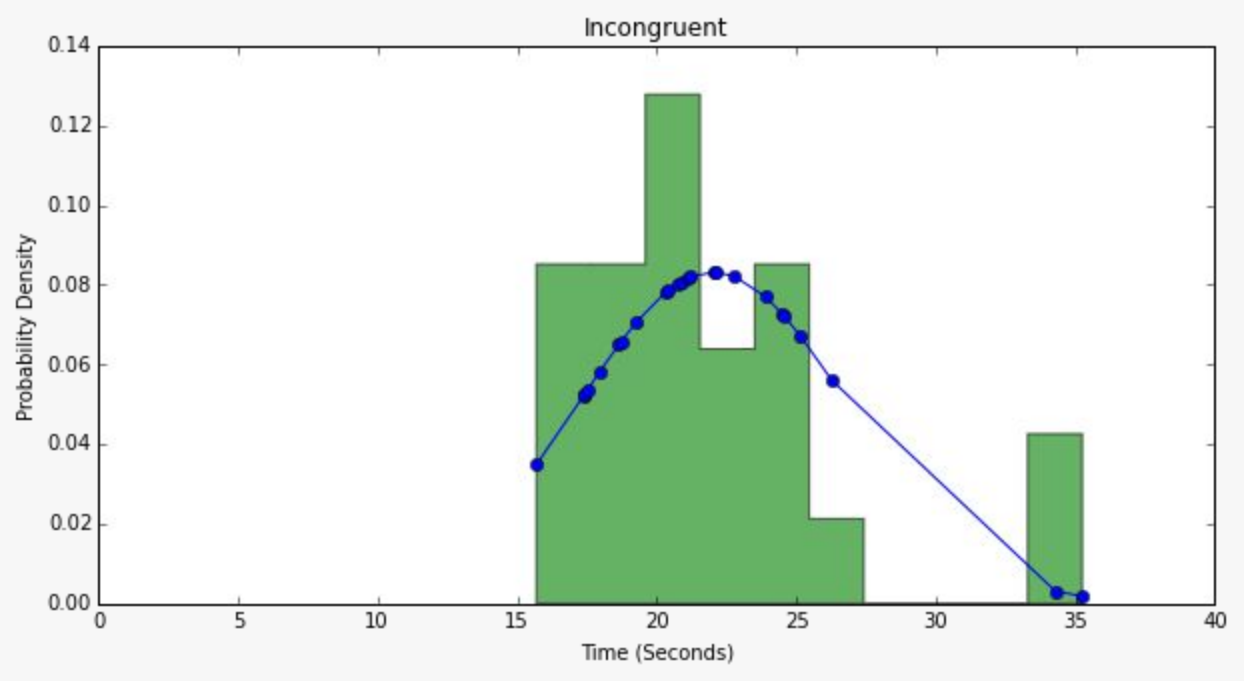
	Congruent	Incongruent	Difference	Percent Diff
count	24	24	24	24
mean	14.0511	22.0159	7.96479	63.4162
median	14.3565	21.0175	7.6665	63.4234
min	8.63	15.687	1.95	9.83518
max	22.328	35.255	21.919	177.209
std	3.55936	4.79706	4.86483	42.3156
se	0.726551	0.979195	0.993029	8.63764

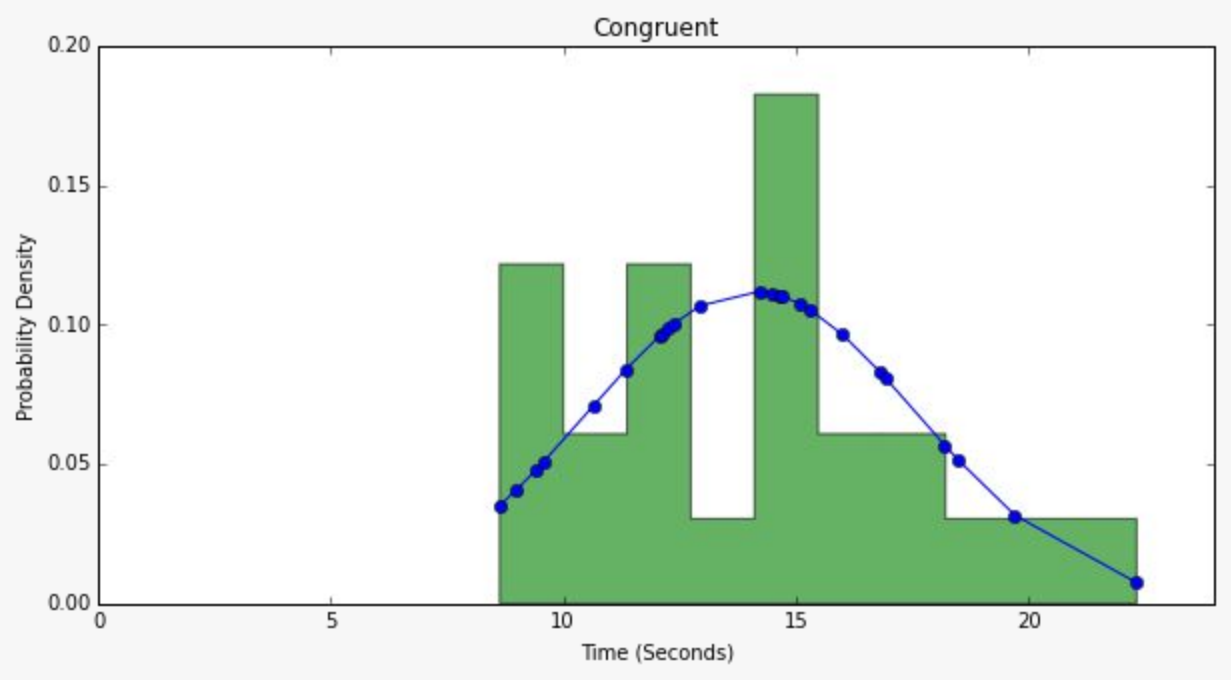
In [65]:

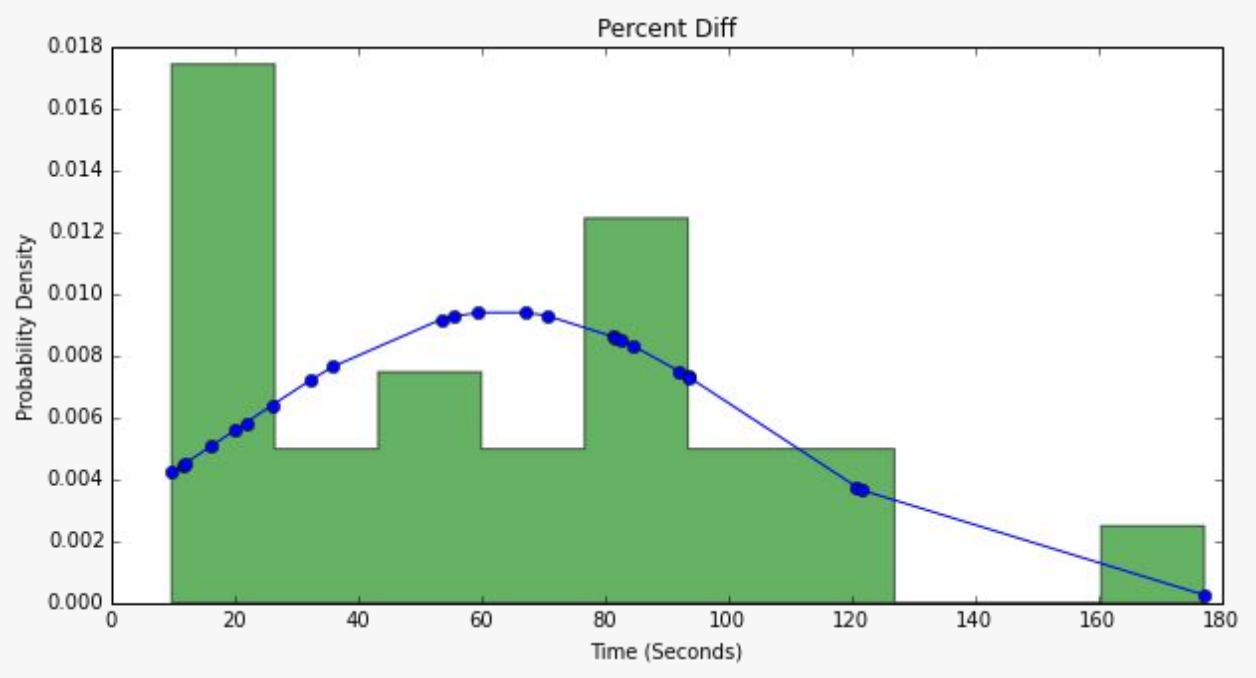
```
def drawPlot(columnData, colIndex, rowIndex1, rowIndex2, title):
    sortedData = sorted(columnData)

    fit = stats.norm.pdf(sortedData, descriptiveStats[colIndex][rowIndex1],
descriptiveStats[colIndex][rowIndex2])  #this is a fitting indeed
    fig, ax = plt.subplots(1,1,figsize=(10,5))
    plt.plot(sortedData,fit,'-o')

    ax.set_xlim(xmin=0)
```

```
    plt.hist(sortedData,normed=True, histtype='stepfilled', alpha=0.6)
    plt.xlabel('Time (Seconds)')
    plt.ylabel('Probability Density')
    plt.title(title)

    return {'fig':fig, 'ax':ax}
```

```
drawPlot(incongruentCol, 'Incongruent', 'mean', 'std', 'Incongruent')
drawPlot(congruentCol, 'Congruent', 'mean', 'std', 'Congruent')
drawPlot(percentDiffCol, 'Percent Diff', 'mean', 'std', 'Percent Diff')
plt.show()
```

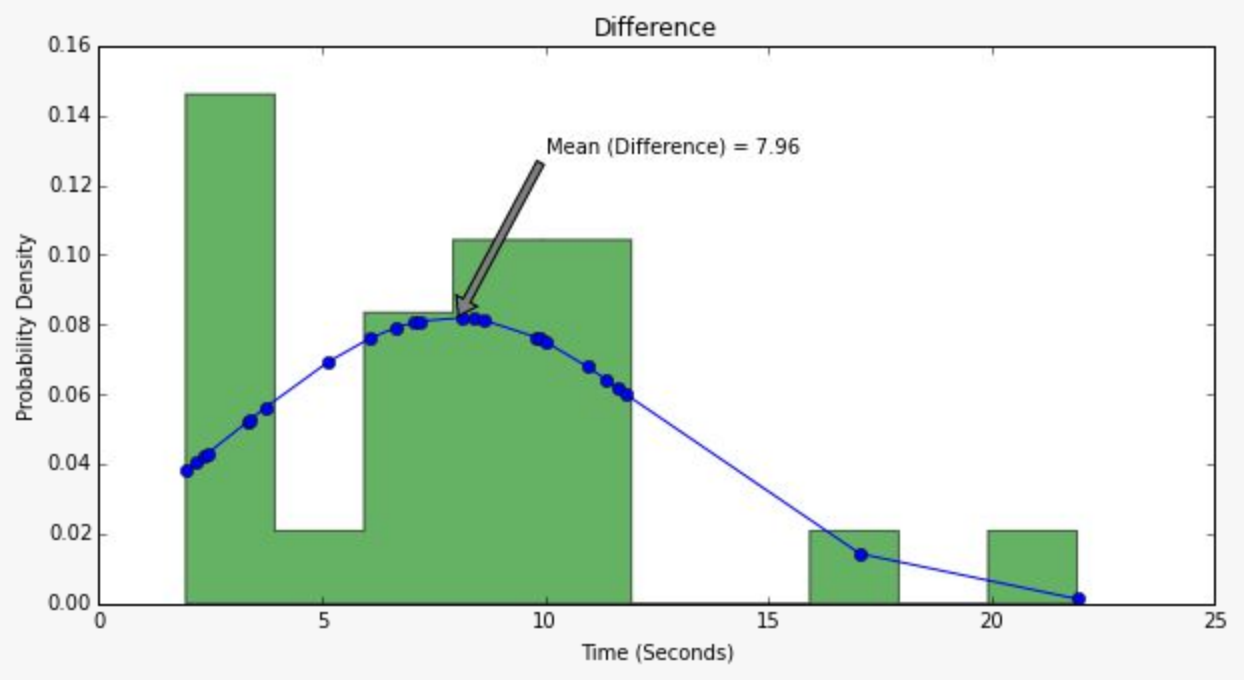






In [70]:

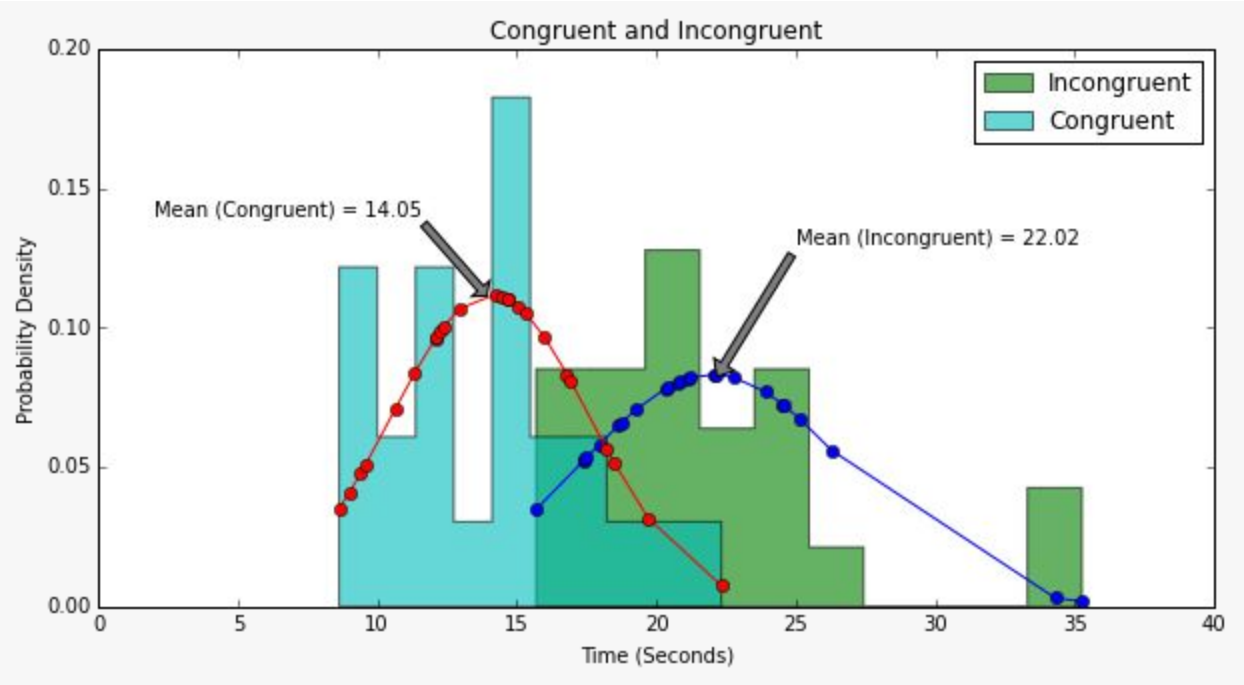
```
result = drawPlot(differenceCol, 'Difference', 'mean', 'std', 'Difference')
result['ax'].annotate('Mean (Difference) = 7.96', xy=(7.96, 0.08), xytext=(10,0.13),
            arrowprops=dict(facecolor='grey', shrink=0.05))
plt.show()
```



In [19]:

```
fig, ax = plt.subplots(1,1,figsize=(10,5))

def drawPlot(columnData, colIndex, rowIndex1, rowIndex2, labelName):
    sortedData = sorted(columnData)

    fit = stats.norm.pdf(sortedData, descriptiveStats[colIndex][rowIndex1],
descriptiveStats[colIndex][rowIndex2])  #this is a fitting indeed

    plt.plot(sortedData,fit,'-o')

    ax.set_xlim(xmin=0)

    plt.hist(sortedData,normed=True, histtype='stepfilled', alpha=0.6, label=labelName)
    plt.xlabel('Time (Seconds)')
    plt.ylabel('Probability Density')

ax.annotate('Mean (Congruent) = 14.05', xy=(14.051, 0.11), xytext=(2.0,0.14),
            arrowprops=dict(facecolor='grey', shrink=0.05))

ax.annotate('Mean (Incongruent) = 22.02', xy=(22.0159, 0.08), xytext=(25.0,0.13),
            arrowprops=dict(facecolor='grey', shrink=0.05))
plt.title('Congruent and Incongruent')
drawPlot(incongruentCol, 'Incongruent', 'mean', 'std', 'Incongruent')
drawPlot(congruentCol, 'Congruent', 'mean', 'std', 'Congruent')
```

```
plt.legend()
plt.show()
```



In [20]:

```
# Calculate the t statistic

tstatistic, pvalue = stats.ttest_rel(incongruentCol,congruentCol)

# TTest returns two tailed pvalue - divide in half for a one tailed test
pvalue = pvalue /2

print "T Statistic = %f" % tstatistic
print "P-Value = %g" % pvalue
```

```
T Statistic = 8.020707
P-Value = 2.0515e-08
```

In [21]:

```
# Cohen's D

cohensD = (descriptiveStats['Incongruent']['mean'] - descriptiveStats['Congruent']['mean']) /
descriptiveStats['Difference']['std']
```

```
print "Cohen's D: %f " % (cohensD)
```

```
Cohen's D: 1.637220
```

In [22]:

```
# Confidence Interval

import math as math

marginOfError = descriptiveStats['Difference']['std']/math.sqrt(descriptiveStats['Difference']['count'])
* 1.714

ciLow = descriptiveStats['Difference']['mean'] - marginOfError
ciHigh = descriptiveStats['Difference']['mean'] + marginOfError

print "Margin of Error: %f" % marginOfError
print "Confidence interval: (%f, infinity)" % (ciLow)
```

```
Margin of Error: 1.702051
Confidence interval: (6.262741, infinity)
```

In [24]:

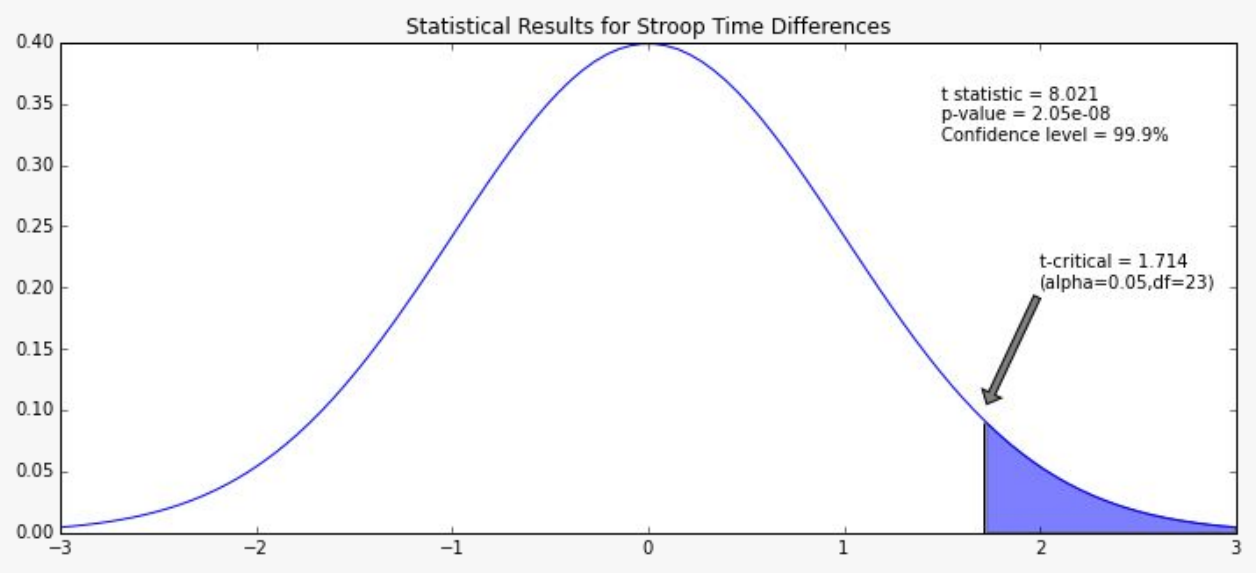
```
# Calculate r squared

tStatisticSquared = tstatistic * tstatistic

rSquared = tStatisticSquared / (tStatisticSquared + 23)

print "r squared: %f" % rSquared
```

```
r squared: 0.736636
```

In [25]:

```
# Display the t-critical and t statistic

fig, ax = plt.subplots(1,1,figsize=(12,5))

mean = 0
variance = 1
sigma = math.sqrt(variance)
x = np.linspace(-3,3,100)
```

```
y = mlab.normpdf(x,mean,sigma)

plt.vlines(1.714, 0, 0.09)

section = np.arange(1.714, 3, 1/20.)
plt.fill_between(x[78:100],y[78:100], facecolor='blue', alpha=0.5)
ax.annotate('t statistic = 8.021\np-value = 2.05e-08\nConfidence level = 99.9%', xy=(1.5, 0.1),
xytext=(1.5,0.32))
ax.annotate('t-critical = 1.714\n(alpha=0.05,df=23)', xy=(1.714, 0.1), xytext=(2.0,0.2),
            arrowprops=dict(facecolor='grey', shrink=0.05),)

plt.title('Statistical Results for Stroop Time Differences')
plt.plot(x,y)

plt.show()
```



In [72]:

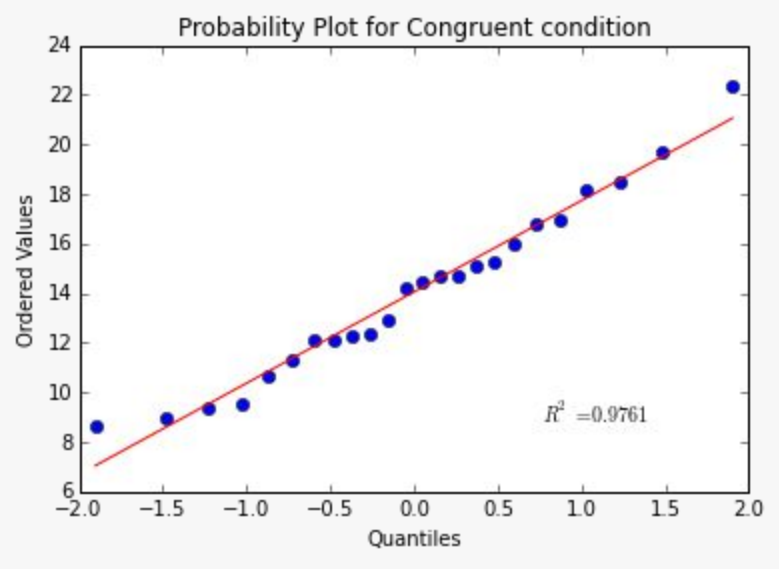
```
# Run the Shaprio-Wilk test

congruentShapiro, congruentP, congruentVals = stats.shapiro(congruentCol, reta=True)

inconShapiro, inconP, inconVals = stats.shapiro(incongruentCol, reta=True)

diffShapiro, diffP, diffVals = stats.shapiro(differenceCol, reta=True)

print "Congruent Shapiro-wilk:   %f, p-value = %f" % (congruentShapiro, congruentP)
print "Incongruent Shapiro-wilk: %f, p-value = %f" % (inconShapiro, inconP)
```

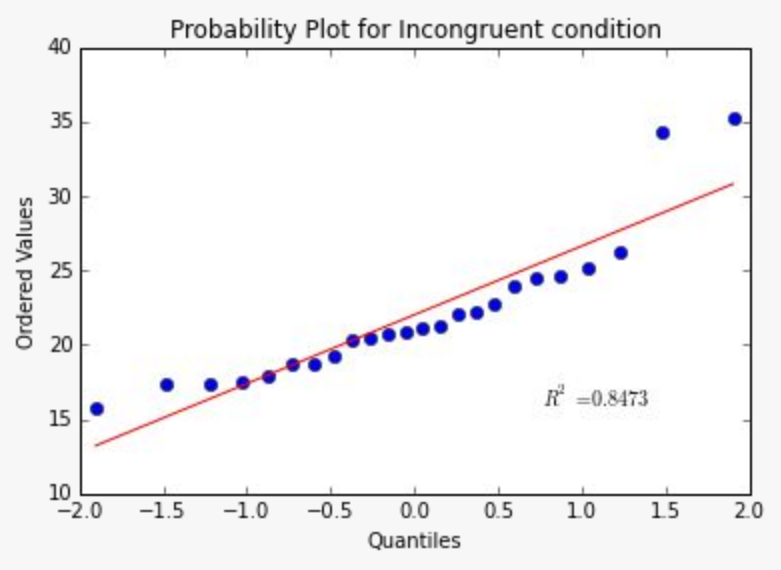
```
print "Difference Shapiro-wilk:  %f, p-value = %f" % (diffShapiro, diffP)
```

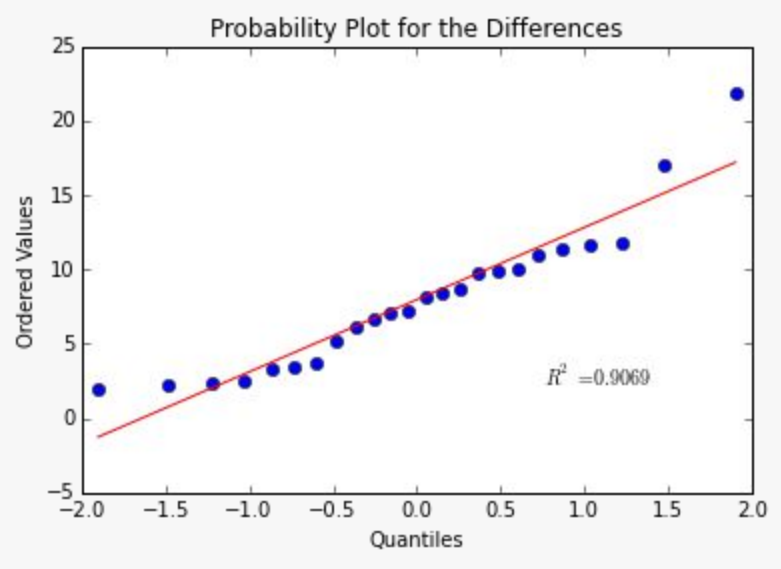
```
Congruent Shapiro-wilk:   0.970923, p-value = 0.689803
Incongruent Shapiro-wilk: 0.853947, p-value = 0.002590
Difference Shapiro-wilk:  0.910420, p-value = 0.036017
```

In [73]:

```
# Draw the QQ plots
stats.probplot(congruentCol, dist="norm", plot=plt)
plt.title("Probability Plot for Congruent condition")
plt.show()

stats.probplot(incongruentCol, dist="norm", plot=plt)
plt.title("Probability Plot for Incongruent condition")
plt.show()

stats.probplot(differenceCol, dist="norm", plot=plt)
plt.title("Probability Plot for the Differences")
plt.show()
```




Probability Plot for Incongruent condition
Ordered Values
Quantiles
R^2 =0.8473


Probability Plot for the Differences
Ordered Values
Quantiles
R^2 =0.9069