Stroop Effect

October 30, 2016

1 Stroop effect

Udacity
Data Science Nano Degree
Project 1
Philipp Vogler
October 2016

1.1 Questions

1.1.1 Q1. What is our independent variable?

Congruency of the words with the color of the ink

1.1.2 What is our dependent variable?

"time it takes to name the ink colors"

1.1.3 Q2a. What is an appropriate set of hypotheses for this task?

The description of the Stroop effect suggests, that it is harder to name incongruent words (ink color is not the word). A harder task should take more time to for fill.

Null - Hypothesis: There **no** significant difference between the time it takes to name congruent words and incongruent words correctly.

Hnull: μincon(response time) = μcon(response time)

Alt.- Hypothesis: There a significant difference between the time it takes to name congruent words and incongruent words correctly.

Halt: μincon(response time) $\neq$ μcon(response time)

1.1.4 Q2b. What kind of statistical test do you expect to perform?

Hypothesis testing with a two-tailed dependent t-test with paired examples.
- While all participants performed both tests, the are dependencies between the samples. - Due to the small number of participants a t-test is perfect over a z-test. - A two-tailed test is appropriate when looking at a difference between two samples

```
In [2]: # Libraries
        import pandas as pd
        import matplotlib.pyplot as plt
```

```
In [3]: # Data import
        path = r'/Users/philipp/Desktop/DS/Projects/Stroop_effect/stroopdata.csv'

        dataFrame = pd.read_csv(path)
        dataFrame
```

Out[3]:

	Congruent	Incongruent
0	12.079	19.278
1	16.791	18.741
2	9.564	21.214
3	8.630	15.687
4	14.669	22.803
5	12.238	20.878
6	14.692	24.572
7	8.987	17.394
8	9.401	20.762
9	14.480	26.282
10	22.328	24.524
11	15.298	18.644
12	15.073	17.510
13	16.929	20.330
14	18.200	35.255
15	12.130	22.158
16	18.495	25.139
17	10.639	20.429
18	11.344	17.425
19	12.369	34.288
20	12.944	23.894
21	14.233	17.960
22	19.710	22.058
23	16.004	21.157

1.1.5 Q3. Descriptive Statistics

In [4]:
```
# Despriptive statistics
dataFrame.describe()
```

Out[4]:

	Congruent	Incongruent
count	24.000000	24.000000
mean	14.051125	22.015917
std	3.559358	4.797057
min	8.630000	15.687000
25%	11.895250	18.716750
50%	14.356500	21.017500
75%	16.200750	24.051500
max	22.328000	35.255000

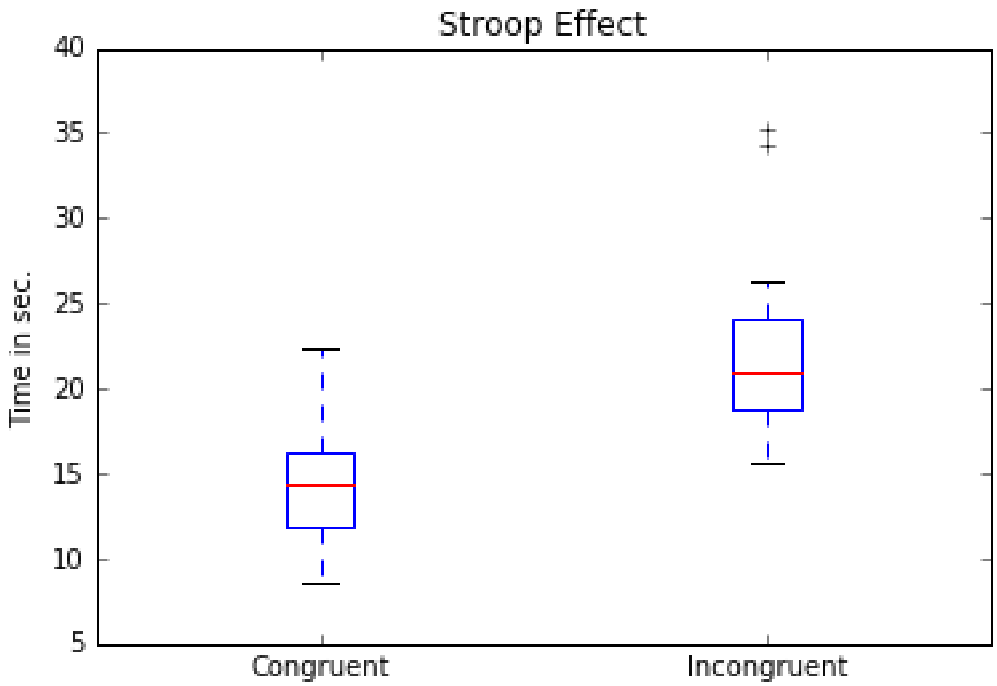
1.1.6 Q4. Plot

In [5]:
```
# Plot
%pylab inline
dataFrame.plot.box()

plt.ylabel("Time in sec.")
plt.title("Stroop Effect")
```

```
Populating the interactive namespace from numpy and matplotlib
```

Out[5]: `<matplotlib.text.Text at 0x114784610>`



While the congruent distribution is skewed toward shorter times, the incongruent distribution is more evenly distributed (outliers notwithstanding). There are two outliers around 35 seconds. The incongruent word condition exercise takes about 5 seconds longer on average.

1.1.7 Q5. Test

Dependend t-test with paird examples

In [6]:
```
# t-Test
#ttost_paired(x1, x2, low, upp[, transform, ...])        test of (non-)equivalence for two depe

# Means
print('Congruent Mean: {}'.format(dataFrame.Congruent.mean()))
print('Incongruent Mean: {} '.format(dataFrame.Incongruent.mean()))

# Point Estimat
point_estimat = dataFrame.Congruent.mean()-dataFrame.Incongruent.mean()
print('Point Estimat: {}'.format(point_estimat))

# Differences
dataFrame['Differences'] = dataFrame.Congruent - dataFrame.Incongruent

# Average Difference
print('Average Difference: {}'.format(dataFrame.Differences.mean()))

# Count
count = dataFrame.Congruent.count()
print('Count: {}'.format(count))
```

```python
    # Degrees of Freedom
    dof = count - 1
    print('Degrees of Freedom: {}'.format(dof))

    # Standard Diviation of Differneces
    std = dataFrame.Differences.std()
    print('Standard Diviation of Differences: {}'.format(std))

    # t-Statistic
    t_statistic = point_estimat / (std / sqrt(count))
    print('t-statistic: {}'.format(t_statistic))

    # Test
    print('Test: t_statistic < t_critical: {}'.format(t_statistic < -2.069))
    print('Test: t_statistic > t_critical: {}'.format(t_statistic > 2.069))
```

```
Congruent Mean: 14.051125
Incongruent Mean: 22.0159166667
Point Estimat: -7.96479166667
Average Difference: -7.96479166667
Count: 24
Degrees of Freedom: 23
Standard Diviation of Differences: 4.86482691036
t-statistic: -8.02070694411
Test: t_statistic < t_critical: True
Test: t_statistic > t_critical: False
```

What is your confidence level and your critical statistic value? alpha = 0.05
CI = 95%
t_critical = 2.069 / -2.069 (from t-table)

Do you reject the null hypothesis or fail to reject it? I reject the null hypothesis.

Did the results match up with your expectations? About 50% difference in means between the samples was evidence for two very different distributions. The t-test confirms this suspicion. The result was expected.

1.2 Sources:

- Wikipedia
- Pandas Documentation
- Matplotlib Documentation
- t-table