

# Paired sample t-test

A health club recently introduced a weight reducing plan. From among a large number of people who participated in the plan, a sample of eleven people was se;ected and their weights were recorded both before and after the completion of plan. The information is given below. Using alpha = 0.05, test whether the diet plan is really effective in reducing weight.

Persons: 1 2 3 4 5 6 7 8 9 10 11

weight(before) in kg: 87.4 92.9 83.6 81.5 89.7 100.5 98.6 88.8 112.4 87.6 92.8

weight(after) in kg: 85.4 88.3 84.7 81.2 83.3 94.6 90.1 87.2 104.6 88.4 91.7

Since the same individuals are involved in two conditions, so the samples are paired. Therefore we shall apply paired sample ttest.

Null hypothesis, H0: Mean difference is zero Alternate hypothesis, H1: mean difference greater than zero

critical value of t at 10 degree of freedom( 11 samples, so 11-1=10) and alpha 0.05 is 1.812

Decision rule = if the calculated t value is greater than critical value then reject H0 (Null hypothesis)

```
In [4]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy
%matplotlib inline
```

```
In [5]: df = pd.read_csv("health club.csv")
df
```

Out[5]:

	Persons	Weight(before) in kg	Weight(after) in kg
0	1	87.4	85.4
1	2	92.9	88.3
2	3	83.6	84.7
3	4	81.5	81.2
4	5	89.7	83.3
5	6	100.5	94.6
6	7	98.6	90.1
7	8	88.8	87.2
8	9	112.4	104.6
9	10	87.6	88.4
10	11	92.8	91.7

```
In [7]: scipy.stats.ttest_rel(df["Weight(before) in kg"], df["Weight(after) in kg"])
```

Out[7]: Ttest\_relResult(statistic=3.160299629287293, pvalue=0.010153633172209735)

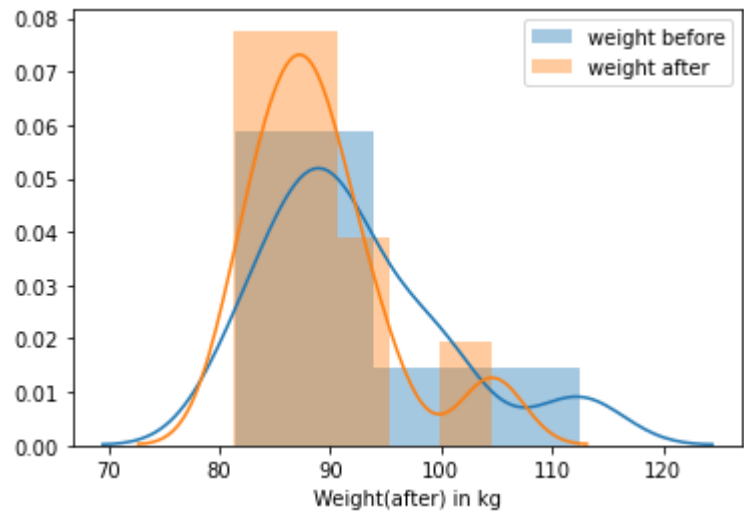
Calculated t value is 3.160 which is greater than critical value of t which is 1.812 at 10 d.o.f at alpha=0.05, so we reject Null Hypothesis and accept Alternate hypothesis.

P value is 0.01 which is less than alpha=0.05 so we reject Null hypothesis and accept the alternate hypothesis which is now statistically significant.

So the weight loss plan is effective.

```
In [8]: sns.distplot(df["Weight(before) in kg"],label="weight before")
sns.distplot(df["Weight(after) in kg"], label="weight after")
plt.legend()
```

Out[8]: <matplotlib.legend.Legend at 0x5cd9e17e48>



```
In [ ]:
```

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy
%matplotlib inline
```

## Two sample t-Test

1) To judge whether there is a difference in the mean incomes of the agents of two insurance companies, a sample is taken from each population and the following results are obtained:

Yearly incomes in Lakhs

Company A 3.2 4.8 5.3 2.2 2.7 6.2 3.4 3.7 4.1 4.2 5.7 2.5

Company B 1.5 2.6 2.8 4.6 3.6 6.2 6.8 1.3 2.1 1.5

Test at alpha = 0.02 whether the difference between mean incomes of two groups is significant. Also calculate p-value and interpret.

Null Hypothesis, H0 : No difference between mean incomes Alternate Hypothesis, H1: significant difference between mean incomes level of significance alpha = 0.02 t\_statistic for 20 d.o.f = 2.528

Decision rule= if calculated t\_statistic is greater than table t\_statistic value then reject Null Hypothesis

```
In [2]: insurance_company = pd.read_csv("insurance company Vohra.csv")
```

```
In [3]: insurance_company
```

Out[3]:

	Company A	Company B
0	3.2	1.5
1	4.8	2.6
2	5.3	2.8
3	2.2	4.6
4	2.7	3.6
5	6.2	6.2
6	3.4	6.8
7	3.7	1.3
8	4.1	2.1
9	4.2	1.5
10	5.7	NaN
11	2.5	NaN

```
In [4]: scipy.stats.ttest_ind(insurance_company.dropna()["Company A"], insurance_company.dropna()["Company B"])
```

```
Out[4]: Ttest_indResult(statistic=0.9278864088271788, pvalue=0.36574256129939575)
```

p value(=.365) is much greater than alpha=0.02, so Null hypothesis cannot be rejected

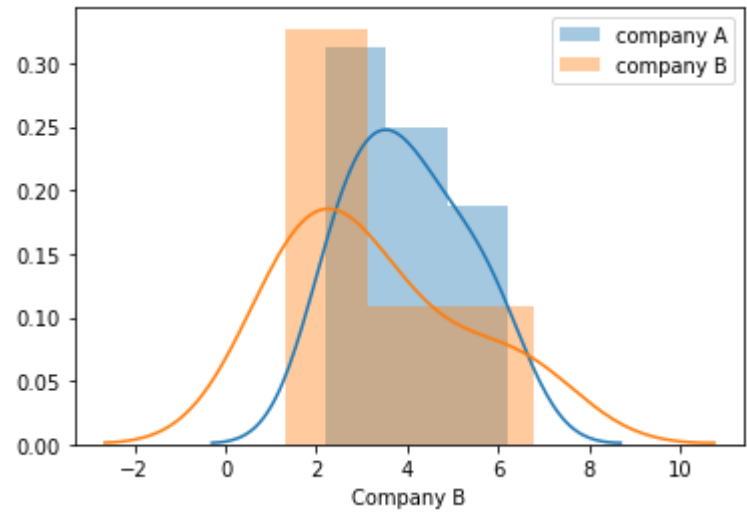
t\_statistic does not exceed critical value of t at alpha=0.02 at 20 d.o.f, so null hypothesis is accepted

here calculated t value is .93 and from table t value is 2.528

there is no difference between mean incomes of the two groups(Null hypothesis accepted)

```
In [5]: sns.distplot(insurance_company["Company A"],label = 'company A')
sns.distplot(insurance_company["Company B"], label = 'company B')
plt.legend()
```

```
Out[5]: <matplotlib.legend.Legend at 0xa768f44d48>
```



```
In [ ]:
```