Artificial Intelligence & Machine Learning Extensions

Sada Narayanappa, PhD

Version 2.0.135

In [ ]:

In [2]:

```
%run 00_basic.ipynb
```

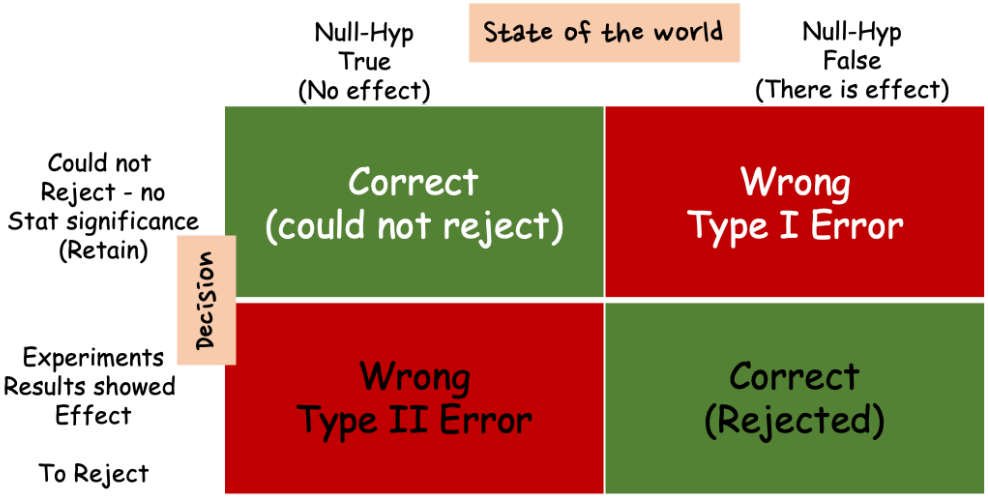
Preliminaries to recall

Type 1 and Type 2 Errors

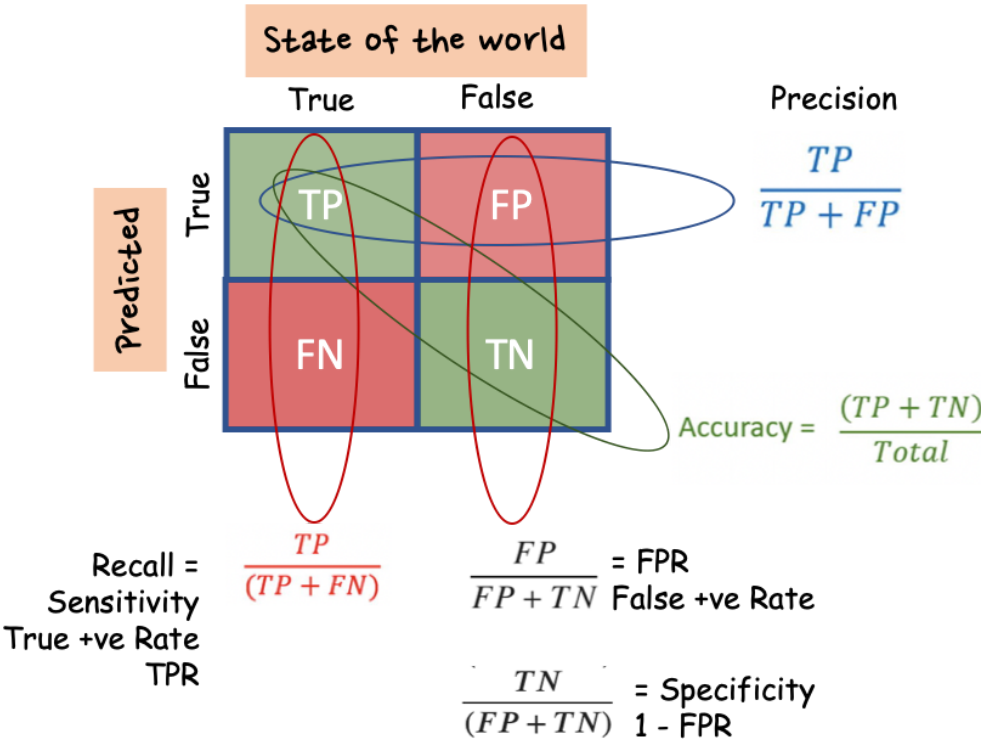
In statistics, when we set up experiments, we state the hypothesis h_0 (AKA null hypothesis) as negative statement that inidicates no relationship to observations and the phenomenon; the reason for stating it as "Null" hypothesus is bit philophical - we can rather prove the absence of the phenomenon. An alternate hypothesis h_1 (which is stated or assumed to be as the opposite of null hypothesis may be true if the experimental results (or evidence) show no significant difference).

When experiments are run, the results are used to evaluate the state of the world. If the experiments are accurate we should be making the correct decisions. Otherwise we commit error:

Type I error is when the null hypothesis is **True**, but it is rejected (False Positive) *Type II* error is when the null hypothesis is **False**, but it could not be rejected (False Negative)



Precision Recall Sensitivity



Precision Among the results predicted as +ve, how many are actually +ve; how precise was the +ve predictions.

$$Precision(+) = \frac{TP}{(TP+FP)}$$

Accuracy is ratio all predictions were correct.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

Recall - how many of real true +ve values were recalled.

$$Sensitivity = Recall(+) = \frac{TP}{(TP+FN)}$$

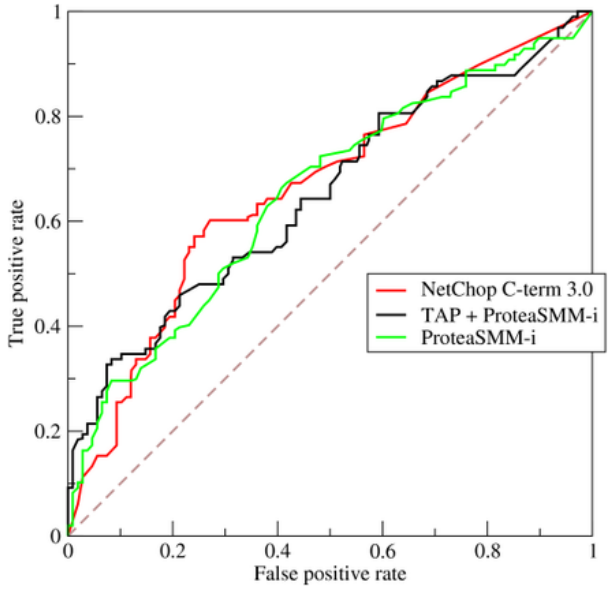
$$Specificity = 1 - FPR = \frac{TN}{(FP+TN)}$$

F1 Score is the harmonic mean of precision and recall.

$$F1_{score} = \frac{2 * precision * recall}{precision + recall}$$

AOC-Area Under the Curve, (ROC- Receiver Operating Characteristic) curve

AOC curve is the plot of TPR Vs FPR for various threshhold settings of parameter of prediction of algorithm.



(Reference Wikipedia)

Ideally we want to have zero False positive rate (FPR) that implies 100% TPR.

AUC of 0.5 says that model is as bad as a random picker.

Hypothesis Testing

Hypothesis testing (or experiement) is done is test the effect of some treatment.

For example: One could test the effect of sleeping pill. Now, if we can make observational study on people using the drug and test if those people are sleeping better. This type os study can show correlation but still it is unclear if the drug is **causing it**.

As we described in the experiemental section, we choose two groups

1. Control
2. Treatment group But the first step is to state the hypothesis

Hypothesis is stated as Null Hypothesis as showing no effect. Null hypothesis H_0.

If we disprove the *Null* hypothesis by showing the probablity of seeing those effects (by observing population parameters) are very low, then we can reject *Null* hypothesis in favor of Alternative hypothesis H_a that states the drug has an effect.

Suppose we know the population mean μ and standard deviation σ;
If we take sample of size n, give them treatment and measure their average $\bar{x}$ and standard deviation = $\frac{\sigma}{\sqrt{n}}$

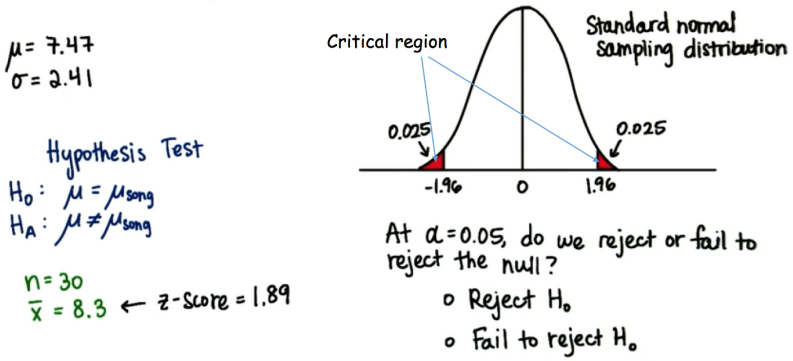
We compute the z-score:

$$z = \frac{\bar{x} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

and test if the z-score is in the critical region.

Here is a complete example.

Concrete Example:



In this example, the population parameters (taken from udacity stats class):

(The actual experiement details does not matter - it is a hypothetical test to test the engagement of student a class if teacher sings of song :))

We want to test the treatment has an effect (positive or negative); therefore we set the alternate hypothesis is $\mu \neq \mu_{song}$; we choose a α level of 5%

Assume that the populative have the following parameters:

$$\mu = 7.47$$
$$\sigma = 2.41$$

Null Hypothesis: H_0 is: SONG HAS NO EFFECT

Alt Hypothesis: H_a is: SONG HAS Some EFFECT (Notice two tailed test)

By running a hypothesis test on a sample of $n = 30$, we found the sample mean

$$\bar{x} = 8.3$$
$$n = 30$$

Question is - does the song had an effect? First, we compute the $Z - score$ and compare to see if it falls in the critical region.

$$Z = \frac{8.3 - 7.47}{\frac{2.41}{\sqrt{30}}} = 1.89$$

All of the following statements are equivalent

- Z is less than -ve z or greater than +ve z
- the p value falls in the critical region

Now, we see that $Z = 1.89$ is not in the critical region.
The probability of getting this result is $1 - .9706 = 0.0294 > 0.025$
We found 0.9706 by looking in Z-table corresponding to Z-score

Therefore, we cannot reject the null hypothesis; In other words, we got this chance.

$H_0: \mu_{song} = \mu$

$H_A: \mu_{song} \neq \mu$

$\mu = 7.47$

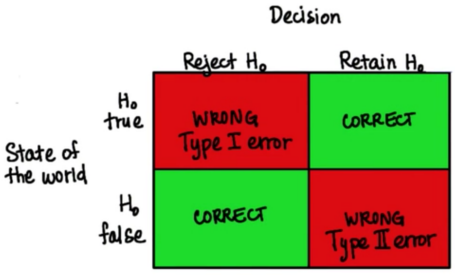
$\sigma = 2.41$

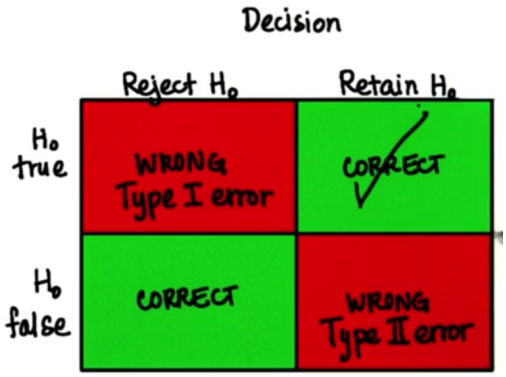
$n = 30 \quad \bar{x} = 8.3$

$\mu_{song} = 7.8$

$\alpha = 0.05$

two-tailed test





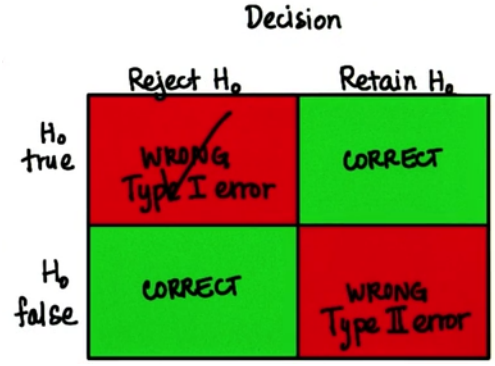
In this example, we got $\bar{x} = 8.3$ with p value was close to 0.025.

Suppose if the true population new $\mu_{song} = 7.8$ then,

$$z = \frac{7.8 - 7.47}{\frac{2.41}{\sqrt{30}}} = .75$$

0.75 is not the critical region, therefore, we correctly retained Null hypothesis;

Although our sample was misleadingly high at 1.89



Now, what if we get the same mean $\bar{x} = 8.3$ with $n = 50$.

Suppose if the true population new $\mu_{song} = 7.8$ then,

$$z = \frac{7.8 - 7.47}{\frac{2.41}{\sqrt{50}}} = .9682$$

0.9682 is not the critical region, therefore, song had no significant effect;

However based on our sample statistic:

$$z = \frac{8.3 - 7.47}{\frac{2.41}{\sqrt{50}}} = 2.43$$

we would reject the Null. making a type I error with probability:

**Type 1 Error* *: H_0 is true - we reject it

**Type 2 Error* *: H_0 is false - we fail to reject it because the sample gave good results

In [1]:

```
import sys
import importlib as imp
if ('Jupytils' in sys.modules):
    reloaded = imp.reload(Jupytils)
else:
    import Jupytils
```

T-test, F-test, Hotelling's T-squared distribution

The t-Test is used to test the null hypothesis that the means of two populations are equal.

$H_0 : \mu_1 - \mu_2 = 0$

$H_1 : \mu_1 - \mu_2 \neq 0$

1. First perform F-test to check if the variances are equal

In this page we don't talk about the Hotelling's T-squared test which is a generalization of t-test for multivariate case.

F-test

Given two series x, y - test if the variances of two population are equal. F-test is easy to conduct. $F - value$ of two series x and y is just the ratio $\frac{Var(x)}{Var(y)}$

H_0 **Null-hypothesis**: Two variances are same

H_a **Research hypothesis**: Two variances are different

F-test is done before t-test to determine if the variance of the two population are different.

Conditions:

- Both random variables are normally distributed
- The samples are independent

If X and Y have a normal distribution, the F-statistic will have F-distribution with $N_x - 1 \; and \; N_y - 1$ degrees of freedom. To define the significance level which corresponds to the value of F-statistic high-precision, F-distribution approximation is used.

####Example:

```
Data: is the number of study hours between male and female

Female: 26, 25, 43,34,18,52
Male:   23,30,18,25,28
```

H_0: Two variances are same H_a: Two variances differ

As you can see from the calculations below: F is 7.373 ($F_{critical}$ = 6.25 , P = 0.03 for one tail which is less than 5% indicated a significant)

$F >> F_{critical}$, therefore we reject null hypothesis and say *"two vairances are different"*

t- test

From t-test we get t=1.47260514049 p=0.187258096865. $p >> p_{critical}(0.05)$ therefore we cannot reject the null and state that there is no difference in the mean study hours between male and female

In [2]:

```
a1 = np.array ( "26, 25, 43,34,18,52,23,30,18,25,28".split(",")).astype(int)
a2 = ["f"]*6 + ["m"]*5;
dfL = pd.DataFrame( {"a1":a1, "a2":a2})

#displayDFs(dfL)
d1 = dfL.loc[dfL['a2'] == 'f']['a1']
d2 = dfL.loc[dfL['a2'] == 'm']['a1']
F = d1.var()/ d2.var();
F_critical = stats.f.ppf(.95,len(d1) - 1, len(d2) - 1);  # n1 - 1, n2-1 are degrees
p = 1- stats.f.cdf(F,5,4)  # 5 and 4 or len(d1)- 1 and len(d2) -1

print (np.array(d1), np.array(d2), "F =", F, "($F_critical,p)=", F_critical, p)
#
# F is >> F_critical and a case for rejecting Null hypothesis
#
equal_variance = True;  # Null Hypothesis
if ( F > F_critical):
    equal_variance = False; # Reject the Null Hypothesis

print ("Equal Variance: ", equal_variance)
```

```
[26 25 43 34 18 52] [23 30 18 25 28] F = 7.373271889400921 ($F_critica
l,p)= 6.25605650216 0.0378883761333
Equal Variance:  False
```

In [3]:

```
# Try other tests for the heck of it
#print stats.bartlett(d2, d1)
#print stats.levene(d2, d1)
#stats.f.pdf(F, d1, d2)
```

In [4]:

```
t_stat, p = stats.ttest_ind(d1, d2, equal_var=equal_variance)

alpha = 0.05;
df = ComputeDegreesOfFreedomFor_t_test(d1,d2,equal_variance);
t_critical_one_tailed=stats.t.ppf(1-alpha, df);
t_critical_two_tailed=stats.t.ppf(1-alpha/2, df);

print (t_critical_one_tailed, t_critical_two_tailed, df)
print( t_stat, p )

print ('''
As p > p_critical of 0.05, we fail to reject the.
The observed difference between the sample means (33 - 24.8)
is not convincing enough to say that the average number of study hours

between female and male students differ significantly.
''')
```

```
1.89457860506 2.36462425101 7
1.47260514049 0.187258096865

As p > p_critical of 0.05, we fail to reject the.
The observed difference between the sample means (33 - 24.8)
is not convincing enough to say that the average number of study hours

between female and male students differ significantly.
```

In [5]:

```
# Another set of examples I found at: https://gist.github.com/mblondel/1761714
# from scipy.stats import ttest_1samp, wilcoxon, ttest_ind, mannwhitneyu
#=========================================================================
# EXAMPLE 1.
# one sample t-test
# null hypothesis: expected value = 7725

# daily intake of energy in kJ for 11 women
daily_intake = np.array([5260,5470,5640,6180,6390,6515,6805,7515,7515,8230,8770])

t_statistic, p_value = stats.ttest_1samp(daily_intake, 7725)


print ("one-sample t-test: p_value = ", p_value , '''
# daily intake of energy in kJ for 11 women
daily_intake = np.array([5260,5470,5640,6180,6390,6515,6805,7515,7515,8230,8770])

# Conducting: one sample t-test
# null hypothesis: expected value = 7725

# p_value < 0.05 => alternative hypothesis:
# data deviate significantly from the hypothesis that the mean
# is 7725 at the 5% level of significance

''')
# one sample wilcoxon-test
wz_statistic, wp_value = stats.wilcoxon(daily_intake - 7725)
print ("one-sample wilcoxon-test", wp_value)
#=========================================================================
# EXAMPLE 2.

# two-sample t-test
# null hypothesis: the two groups have the same mean
# this test calls F test to check if equal variance can be assumed
# independent groups: e.g., how boys and girls fare at an exam
# dependent groups: e.g., how the same class fare at 2 different exams

energ = np.array([
# energy expenditure in mJ and stature (0=obese, 1=lean)
[9.21, 0],[7.53, 1],[7.48, 1],[8.08, 1],[8.09, 1],[10.15,1],[8.40, 1],[10.88, 1],
[6.13, 1],[7.90, 1],[11.51,0],[12.79,0],[7.05, 1],[11.85,0],[9.97, 0],[7.48,  1],
[8.79, 0],[9.69, 0],[9.68, 0],[7.58, 1],[9.19, 0],[8.11, 1]]
)

# similar to expend ~ stature in R
group1 = energ[:, 1] == 0
group1 = energ[group1][:, 0]
group2 = energ[:, 1] == 1
group2 = energ[group2][:, 0]
(equal_variance, F , F_c , p,d) = Ftest(group1, group2)
t_statistic, p_value = stats.ttest_ind(group1, group2,  equal_var=equal_variance)

# p_value (0.00079) < 0.05 => alt hypothesis: Mean value differ 5% significance leve
print ("========================\nExample 2: two-sample t-test, p_value=", p_value)

# two-sample wilcoxon test
```

```
# a.k.a Mann Whitney U
u, p_value = stats.mannwhitneyu(group1, group2)
print ("two-sample wilcoxon-test p_value=", p_value)

#=============================================================================
# EXAMPLE 3
# pre and post-menstrual energy intake
intake = np.array([
[5260, 3910],[5470, 4220],[5640, 3885],[6180, 5160],[6390, 5645],[6515, 4680],
[6805, 5265],[7515, 5975],[7515, 6790],[8230, 6900],[8770, 7335],
])

pre = intake[:, 0]
post= intake[:, 1]

# paired t-test: doing two measurments on the same experimental unit
# e.g., before and after a treatment
t_statistic, p_value = stats.ttest_1samp(post - pre, 0)

# p < 0.05 => alternative hypothesis:
# the difference in mean is not equal to 0
print ("========================\nExample 3: paired t-test p_value=", p_value)

# alternative to paired t-test when data has an ordinary scale or when not
# normally distributed
z_statistic, p_value = stats.wilcoxon(post - pre)

print ("paired wilcoxon-test p_value=", p_value)
```

```
one-sample t-test: p_value =  0.0181372351761
# daily intake of energy in kJ for 11 women
daily_intake = np.array([5260,5470,5640,6180,6390,6515,6805,7515,7515,
8230,8770])

# Conducting: one sample t-test
# null hypothesis: expected value = 7725

# p_value < 0.05 => alternative hypothesis:
# data deviate significantly from the hypothesis that the mean
# is 7725 at the 5% level of significance


one-sample wilcoxon-test 0.0261571823293
========================
Example 2: two-sample t-test, p_value= 0.00079899821117
two-sample wilcoxon-test p_value= 0.00212161338588
========================
Example 3: paired t-test p_value= 3.05902094293e-07
paired wilcoxon-test p_value= 0.00333001391175
```

In [17]:

```
#%run  "../StatUtils.py"
(eq,a,b,c,d) = Ftest(group1, group2)
(eq,a,b,c,d)
```

Out[17]:

```
(True,
 1.2275707804649485,
 2.848565142067682,
 0.3612456947013797,
 'Equal Variance: True, F= 1.2275707804649485, F_critical=2.8485651420
67682,P=0.3612456947013797')
```

In [ ]:

```

```

In [1]:

```
import sys
import importlib as imp
if ('Jupytils' in sys.modules):
    reloaded = imp.reload(Jupytils)
else:
    import Jupytils
```

Chi Square test

Chi-square (χ^2) test is used to test null-hypothesis if categorical observed values are same as expected frequencies. The following conditions are met:

- The variable under study is categorical.
- The expected value of the number of sample observations in each level of the variable is at least 5.

This approach consists of four steps: 1) state the hypotheses, 2) formulate an analysis plan, 3) analyze sample data, and 4) interpret results.

Chi-square (χ^2) test is used on two scenarios

- **Goodness of FIT**: How observed frequency fits the expected frequency
- **Test for Independence**: If variables have no influence on each other (i.e. they are independent)

More example to follow

REFERENCES:

- http://math.hws.edu/javamath/ryan/ChiSquare.html (http://math.hws.edu/javamath/ryan/ChiSquare.html)

Goodness of FIT Example

Problem: In order to climb Mount Chesta we need to make some decisions on hiring a guide company.

A guide company claims that 41% succesful attempts last year. It is also known that 33% of 15,000 attempts each year are succefful.

Summit up: 41% succesful attempts of 100 attempt
Generally: 33% succesful attempts of 15,000 attempt

	Successful	unsuccessful	Totals
Expected	33	67	100
Observed	41	59	100
Totals	74	126	200

We want to check if we have a better chance of climbing by hiring the guide company.

We want to check how well the observed proportions fit the population proportions (or expected proportions).!

H0 (Null): expected == observed (i.e. no difference)

H1 (Alternative): Null hypothesis is not true

CHI-square

$\chi^2 = \sum_{k=1}^{n} \frac{(f_o - f_e)^2}{f_e}$

χ^2 Value is smaller when observed value is closer to the expected value.

$\frac{(41-33)^2}{33} + \frac{(59-67)^2}{67} = 2.89$

Degrees of Freedom (df) = 1; (This will be the number of categories minus one)

$p = 0.0889 > 0.05$ - we fail to reject the null-hypothesis and state that *guide company* is not worth hiring. There is about 9% chance we could be wrong and these differences are due to chance

In [2]:

```
A = stats.chisquare([41,59] , [33,67])
Ho = "FAIL TO REJECT: *There is no difference: we fail to reject null hypothesis"
Chi2_c = stats.chi2.ppf(.95, 1);
if (A[1] < 0.05):
    Ho = "We can reject Ho: there is a difference"
print (A, "\n", Ho, "\nChi2 Critical: ", Chi2_c, " Chi2 Value: ", A[0])
print (1-stats.chi2.cdf(2.89,1))
```

```
Power_divergenceResult(statistic=2.8946178199909545, pvalue=0.08887585044058065)
 FAIL TO REJECT: *There is no difference: we fail to reject null hypothesis
Chi2 Critical:  3.841458820694124  Chi2 Value:  2.8946178199909545
0.08913092551708635
```

Test for independece

Chi-square (χ^2) for independece whether or not "two variabled are independent".

Chi-square test for **goodness of FIT** captures data for in two rows. For example: *Which season do you prefer*, in that case, we may have 4 responses *"Summer, Winter, Spring, Fall"

	Response1	**Response2**	**Response3**
Treatment1			
Treatment2			

-

Chi-square (χ^2) also helps for *independece* whether or not "two variables are independent". For example, in above table, we want to know if Response1, Response2, Response3 are independent of Treatment1 or Treatment 2.

In this case, instead of having observed and expected in two rows, we will looking at number of participants (or subjects) fall into variable1 and variable2.

Example

Problem: We want to know if wording of a question influences the answer.

In an experiement, 150 students from University of Washington were shown one minute film clip of a car accident. The students were separated into three equal groups each student chosen randomly into any of the three group and thus each group consisting of 50 students. After the clip was shown:

> **Group 1** was asked : *How fast were the cars going when they *hit** each other

> **Group 2** was asked : *How fast were the cars going when they *smashed into** each other

> **Group 3** was asked : *How fast were the cars going* (No question about the speed of the car)

After one week, all students were asked *"Did you see any broken glass"* and the responses were noted as below

> - Group1 *: responded 7 out of 50 answered yes!
> - Group2 *: responded 16 out of 50 answered yes!
> - Group3 *: responded 6 out of 50 answered yes!

This is summarized in the following table:

	HIT	smashed	Control	Total
YES	7	16	6	29
NO	43	34	44	121
TOTAL	50	50	50	150

In this case, 29/150 said *YES*, 121/150 said *NO*. The expected "YES" response is thus $\frac{29}{150}$ for "YES" group; $\frac{121}{150}$ for "NO" group.

Now we can add the expected response to each cell. In HIT group consisting of 50 students, we expect $\frac{29}{150} * 50 = 9.67$ to say YES (and same is true for each group since there are same number of students in each group). Similarly we expect $\frac{121}{150} * 50 = 40.33$ to say "NO" in each group. Lets update the table with the expected values as follows:

-

	HIT	smashed	Control	Total
YES	7 **9.67**	16 **9.67**	6 **9.67**	29
NO	43 **40.33**	34 **40.33**	44 **40.33**	121
TOTAL	**50**	50	50	150

$$\chi^2 = \sum_{k=1}^{n} \frac{(f_o - f_e)^2}{f_e}$$

χ^2 Value is smaller when observed value is closer to the expected value.

$$\frac{(7-9.67)^2}{9.67} + \frac{(16-9.67)^2}{9.67} + \frac{(6-9.67)^2}{9.67} + \frac{(43-40.33)^2}{40.33} + \frac{(34-40.33)^2}{40.33} + \frac{(44-40.33)^2}{40.33} = 7.7779$$

Degrees of Freedom is 2 ((number of groups -1) * (number of responses -1))

From the Chi-square table: https://people.richland.edu/james/lecture/m170/tbl-chi.html (https://people.richland.edu/james/lecture/m170/tbl-chi.html)
we see the critical value is 5.991. Now Chi2 is 7.777 >> 5.9991 and p-value as calculated from the program below is 0.02 << 0.05.

We reject the null-hypothesis and state that the "**word**" "*SMASHED*" had a difference!

####strength of the relationship:

When we have a contigency table greater than 2x2, we can use * Cramer's V* $(\phi_c) = \sqrt{\frac{\chi^2}{n(k-1)}}$ Where K is the smaller of number of rows or columns: in this case, it is number of rows = 2; n is the total number of subjects = 100

Therefore * Cramer's V* $(\phi_c) = \sqrt{\frac{7.77^2}{100(2-1)}} = .227$

From Cramers V table: for k-1 = 1 indicates small effect!

In [3]:

```
chi2, p, ddof, expected = stats.chi2_contingency( [[7,16,6], [43,34,44]] )
Ho = "KEEP Null Hyp: There is no difference: we fail to reject null hypothesis"
if (p < 0.05):
    Ho = "REJECT H0: there is an effect"

Chi2_c = stats.chi2.ppf(.95, 2);
n = 150;
k = 2;
cramers_v = sqrt(chi2/ ( n * (k-1)))
print ("Chi stat: ", chi2, " Chi Critical: ", Chi2_c, "\np=", p, " ddof: ", ddof, " expected:\n",

# Another way to compute p using CDF
print ("Finding p another way: ", 1-stats.chi2.cdf(chi2,2))

print (" Cramers V: ", cramers_v)
```

```
Chi stat:  7.779994300370478  Chi Critical:  5.991464547107979
p= 0.02044540430346961  ddof:  2  expected:
 [[ 9.67  9.67  9.67]
 [40.33 40.33 40.33]]
 REJECT H0: there is an effect
Finding p another way:  0.02044540430346964
 Cramers V:  0.22774246127838463
```

There are 110 houses in a particular neighborhood.

- Liberals live in 25 of them,
- moderates in 55 of them, and
- conservatives in the remaining 30.

An airplane carrying 65 lb. sacks of flour passes over the neighborhood. For some reason, 20 sacks fall from the plane, each miraculously slamming through the roof of a different house. None hit the yards or the street, or land in trees, or anything like that. Each one slams through a roof. Anyway, 2 slam through a liberal roof, 15 slam through a moderate roof, and 3 slam through a conservative roof.

Null Hypothesis: Sacks of flour hit houses at random?

Should we reject the hypothesis?

-

Given the numbers of liberals, moderates and conservative households, we can calculate the expected number of sacks of flour to crash through each category of house:

20 sacks x 25/110 = 4.55 liberal roofs smashed
20 sacks x 55/110 = 10.00 moderate roofs smashed
20 sacks x 30/110 = 5.45 conservative roofs smashed
Set up the table for the goodness-of-fit test:

Category	Observed	Expected	Obs-Exp	(Obs-Exp)^2 / Exp
Liberal	2	4.55	-2.55	1.43
Moderate	15	10.00	5.00	2.50
Conservative	3	5.45	-2.45	1.10
Total	20	20.00	0	5.03

In a simple test like this, where there are three categories and where the expected values are not influenced by the observed values, there are two degrees of freedom. Checking the table of critical values of the chi-square distribution for 2 d.f., we find that 0.05 < p.

That is, there is greater than a 5% probability of getting at least this much departure be tween observed and expected results by chance.
Therefore, while it appears that moderates have had worse luck than liberals and conservat ives, we **CANNOT REJECT** the NULL hypothesis - which means sacks struck houses at rando m.

In [4]:

```
observed= [2,15,3]
expected=[25*20./110,55.*20/110,30.*20/110 ]

Chi2_c = stats.chi2.ppf(.95, 2);

A = stats.chisquare(observed, expected)
print ("Chi Critical: ", Chi2_c, " Ch2: ", A[0], " p-value: ", A[1])
print ("Observed/Expected: ", (observed, expected))
```

```
Chi Critical:  5.991464547107979  Ch2:  5.03  p-value:  0.08086291220670366
Observed/Expected:  ([2, 15, 3], [4.545454545454546, 10.0, 5.454545454545454])
```

Another Example

Example from: http://stattrek.com/chi-square-test/independence.aspx?Tutorial=AP

	Voting Preferences			Row total
	Republican	Democrat	Independent	
Male	200	150	50	400
Female	250	300	50	600
Column total	450	450	100	1000

H0: Gender and voting preferences are independent. Ha: Gender and voting preferences are not independent.

-

Interpret results. Since the P-value (0.0003) is less than the significance level (0.05), we cannot accept the null hypothesis. Thus, we conclude that there is a relationship between gender and voting preference.

In [5]:

```
house = [ [ 200,150,50], [ 250,300,50 ] ]
chi2, p, ddof, expected = stats.chi2_contingency( house )
msg = "Test Statistic: {}\np-value: {}\nDegrees of Freedom: {}\n"
print( msg.format( chi2, p, ddof ) )
print( expected )
```

```
Test Statistic: 16.203703703703702
p-value: 0.0003029775487145488
Degrees of Freedom: 2

[[180. 180.  40.]
 [270. 270.  60.]]
```

In [ ]:

```

```

In [17]:

```
import sys
import importlib as imp
if ('Jupytils' in sys.modules):
    reloaded = imp.reload(Jupytils)
else:
    import Jupytils
```

Mann Whitney test (also called the Mann-Whitney-Wilcoxon (MWW))

is a nonparametric test of the null hypothesis that two samples come from the same population against an alternative hypothesis, especially that a particular population tends to have larger values than the other.

- Non parametric test used when data comes from non-normal distribution
- Can be used with small samples

There are some situations when it is clear that the outcome does not follow a normal distribution. These include situations:

- when the outcome is an ordinal variable or a rank,
- when there are definite outliers or
- when the outcome has clear limits of detection

> **Use** : To compare a continuous outcome in two independent samples $group_1$ and $group_2$.

> **Null Hypothesis** : H0: Two populations are equal

> **Test Statistic** : The test statistic is U, the smaller of n_1, n_2 is the number of entries in group1 and group2

$U = min(U_1, U_2)$

$U_1 = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1$

$U_2 = n_1 n_2 + \frac{n_2(n_2+1)}{2} - R_2$

where R_1 and R_2 are the sums of the ranks in $group_1$ and $group_2$, respectively.

Decision Rule: Reject H_0 if $U <$ critical value from table in favor of H_a the research hypothesis

References:

- http://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Nonparametric/BS704_Nonparametric4.html (http://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Nonparametric/BS704_Nonparametric4.html)

-

####NOTE: If the sample size is at least 20, then one could use Z-values to test

(Reminder z for two tailed z test is 1.96 and for one tailed it is 1.65 I think)

If $Z_{calculated}$ is less than -1.96 or greater than +1.96, we reject the Null hypothesis"

Example

```
Following example is taken from you tube video
https://www.youtube.com/watch?v=hw3z49QoB1s

Data:
Data is a list of "Scores" obtained in an exam by two groups who were sressed and not-stre
ssed.
Question, is these a difference between these groups?

Y Stressed Y:    44    50    68    70    72    75    76    81    83    88    92    94
No Stress  N:    74    78    79    82    87    90    91    92    92    93

H0: There is no difference in scores between Stress and no-stress Groups
Ha: There is a difference

Test: 2 sided (because of some difference)
n1 = 12
n2 = 10
U critical (from table): 29
Result: if U is less than 29 we reject H0 in favor of Ha (i.e there is difference)

PDF: http://ocw.umb.edu/psychology/psych-270/other-materials/RelativeResourceManager.pdf

From the calculations below, we find:
Z = 1.84626532551 p = 0.129 (Note we multiply by 2 for two sided p-value)
U = 32.0 p = 0.0694  (Note we multiply by 2 for two sided p-value)

32 is not less than 29, therefore we fail to reject the H0.
(Also the p > 0.05, Z value is within -1.96 and +1.96 - all unabling to reject H_0)
i.e. There is no difference in scores between Stress and No-Stress groups
```

-

In [16]:

```
fileName="data/mann-whitney-test1.csv"

dfL  = LoadDataSet(fileName, columns=None);
displayDFs([dfL])
d1 = dfL.loc[dfL['Stress'] == 'N']['Score']
d2 = dfL.loc[dfL['Stress'] == 'Y']['Score']

z_stat1, p_val1 = stats.ranksums(d1, d2)
u, p_val2 = stats.mannwhitneyu(d1, d2,1)
print ('''
Mann Whitney fails with large values of P
''',d1.shape, d2.shape, "\nU-statistic: ",z_stat1, " P value: ", p_val1 , "\nMWW U stat: ", u, " P
```

22x2 var: DFF_PY_VAR_tableID_1579399727219

	Score	Stress
0	44	Y
1	50	Y
2	68	Y
3	70	Y
4	72	Y
5	74	N

<< >> Save ↻ Search>

```
Mann Whitney fails with large values of P
 (10,) (12,)
U-statistic:  1.8462653255082035  P value:  0.06485368983936193
MWW U stat:  32.0  P:  0.06947051710021464
```

```
##Another example
A physician is interested in the effect of an anaesthetic on reaction times. Two groups are
compared,
    * Group A taking anaesthetic
    * Group B without taking the anaesthetic.

Subjects had to react on a simple visual stimulus. Reaction times are not normally distributed
in this experiment, so data is analysed with the Mann-Whitney U-Test for ordinal scaled
measurements. The table below shows the rank-ordered data:

####Example taken from:
* Example From https://secure.brightstat.com/index.php?p=c&d=1&c=2&i=5
```

-

* Look at the results:
https://secure.brightstat.com/img/content/npartests/UTest/ex/Example_MWU.pdf

H0: There is no difference in reaction times in groups taking anaesthetic or not
Ha: There is a difference

Test: 1 sided (We want Anaesthetic group to be slower) at 5% confidence
n1 = 14
n2 = 12
U critical (from table): 51 (Fro two tailed it is 45)
Result: if U is less than 51 (less than 45) we reject H0 in favor of Ha (i.e there is difference)

PDF: http://ocw.umb.edu/psychology/psych-270/other-materials/RelativeResourceManager.pdf

From the calculations below, we find:
Z = -2.16 p = 0.0307 (Note we multiply by 2 for two sided p-value)
U = 42.0 p = 0.0163 (Note we multiply by 2 for two sided p-value)

42 is less than 51, therefore we reject the H0.
(Also the p < 0.05, Z value is outside of 1.65 (or for 2 tailed -1.96 and +1.96) - all reject H_0)
i.e. There is a difference The anaesthetic group shows significantly slower reaction times than the non-anaesthetic group

-

In [14]:

```
fileName="data/mann-whitney-test2.csv"

dfL  = LoadDataSet(fileName, columns=None, comment='#');
d2 = dfL.loc[dfL['Group'] == 'A']['Mean']
d1 = dfL.loc[dfL['Group'] == 'B']['Mean']


z_stat1, p_val1 = stats.ranksums(d1, d2)
u, p_val2 = stats.mannwhitneyu(d1, d2,1)


print( '''
Mann Whitney fails with large values of P
''',"n1: ", d1.shape, " n2:", d2.shape, "\nRank Sums: z: ",z_stat1, " p: ", p_val1 , "\nMann-Whitr

displayDFs(dfL);
```

```
Mann Whitney fails with large values of P
 n1:  (12,)  n2: (14,)
Rank Sums: z:  -2.1602468994692865  p:  0.03075356125927459
Mann-Whitneyt U:  42.0  p:  0.01632518745228646
```

26x2 var: DFF_PY_VAR_tableID_1579399714034

	Mean	Group
0	131	B
1	135	A
2	138	B
3	138	B
4	139	A
5	141	B

<< >> Save ↻ Search>

-

In [5]:

```
# Example from: https://www.youtube.com/watch?v=nRAAAp1Bgnw
#
#
s1= [28,31,36,35,32,33,12,18,19,14,20,19]
s2= "a,a,a,a,a,a,b,b,b,b,b,b".split(",");
dfL = pd.DataFrame( {"Data":s1, "Group":s2})

displayDFs(dfL)
d2 = dfL.loc[dfL['Group'] == 'a']['Data']
d1 = dfL.loc[dfL['Group'] == 'b']['Data']


z_stat1, p_val1 = stats.ranksums(d1, d2)
u, p_val2 = stats.mannwhitneyu(d1, d2,1)
print ('''
Mann Whitney fails with large values of P
''',d1.shape, d2.shape, "\n",z_stat1, p_val1*2 , "\n", u, p_val2 * 2)
```

12x2 var: DFF_PY_VAR_tableID_1579399452960

	Data	Group
0	28	a
1	31	a
2	36	a
3	35	a
4	32	a
5	33	a

<< >> Save ↻ Search>

```
Mann Whitney fails with large values of P
 (6,) (6,)
 -2.8823067684915684 0.007895503713806915
 0.0 0.004998124765082452
```

In [ ]:

```

```

-

In [11]:

```
%run 00_basic.ipynb
```

What is an Anomaly

Anomaly is something that is not normally observed. It does not mean fault or failure. Just something different.

A **fault** in a system is a negative behaviour and has the potential to cause **Failure** which is the inability of the system to execute the intended operation.

Anomalies are also referred to as discordants, deviations, outliers, novelty - something that stands out when compared to normal population.

Hawkins [1] defines:

> "An outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism."

Applications of Anomaly Detection

System operations is to detemine if the system is operating normally or were there any deviations from is nomrla beharviour. For example a spacecraft operations can be monitored using a system in place.

Intrusion Detection Systems : is to check the behavior of system calls, Network traffic, activity during normal and holidays are not unsual. An unusual slowness or elevated activitied may indicate an attack

In **Fraud Detection** a typical usage pattern and buyng behavior of the individuals can be modelled and compared with each future transaction to detect an anomalous transaction. The model can take into account the charge amount, location, speed and frequency of purchases etc. to compare to detect normal or fradualant transactions.

Medical Diagnosis, the authors have applied to predict Ashtma triggers in patients

Weather predictions or **aging predictions** to see if there are unsual patterns that lead to environmental trends.

Determine and set normal Operating range of values for a measured entity. For example, in spacecraft operations one could ask, what is the normal operating temperature of thermal heaters. In medical application, one migth want to know normal heart rate during rest and activity periods among different segments of the population.

Other applications one can imagine are **Cyber Security Money Laundering Banking and Financial applications** etc.

Concerns of Anomaly Detection System

Challenges, Methods and Approaches to Anomaly Detection Algorithms

Anomaly is out of the norm behaviour, therefore usually due to inherant nature of it, one might often enounter:

- significant class imbalance.
- Concept Drift - the behavior evolves and drifts its dynamic
- New Anomalies - it is unlikely to enumerate all the anomalies
- Lack of supervised set (especially during new product development)
- Class Overlap - anamlous data can overlap on non-anomalous datasets.
- Anomaly data in very high dimenstions - the spacecraft data sets range from 5000 to 150000 sensors

Methods

The methods to detect Anomaly could be

> - Supervised
> - Unsupervised
> - Semi Supervised

- Denisity Based methods (AKS Procimity based Methods)

  > - DBSAN -LOF

- Distance Baed

  > - Clustering
  >
  >   > - K-NN
  >   > - K-Means
  >
  > - Regression Hyperplane distane

- Parametric

  > - Gaussian mixture Models
  > - Single class SVMs
  > - Extreme Value Theorem
  > - Hidden Markov Models
  > - Isolation Forests
  > - Extreme Value Analysis
  > - Linear Models
  > - Spectral Models

- System Invariance Models

  > - System Invariant Analysis Technology (SIAT) [2][3]

- Deep Neural networks (numerous articles)

    - Sequence models
    - Long Short Term Memory (LSTM)
    - Convolutional Neural networks (CNN)

Short sight on Detection and need for extentions'

Almost all methods consists of developing a model by observing the data in normal time. Compare the model to remaining dataset and classify the new observations as abnormal or normal.

It is just not sufficient to detect an anomaly. The complex models such as LSTM's can detect anomalies, however just merely detecting anomaly does not solve real world issues.

The other concerns are:

- When is the time to retrain the model? or how long does a trained model can predict anomalies - model warranty
- **Most critical** are how to explain the anomaly; the cause of the anomaly; It is importance to note that the LSTM models can detect anomalies with high fidelity. However it is very **critical** to explain the anomaly; most methods fail to explain the cause. the models such as SIAT are built to show the cause of anomaly and trouble shoot the root cause and offer remedial recommendations.
- How to capture and visualize the **Logical mapping of the system**. SIAT models inherently show the dynamics or logical-model of the system. This is especially important to compare the constllation of similar systems. In case of satellites, how does constellation of spacecrafts behave - an important dynamics to understand the space weather effecs on spacecraft operations that undergo frequent **Elecro static discharge (ESD)** events that cause what is known as **Single Event Failures SEF** that causes intermittent failure of the spacecraft; these intermittent failures have tremondous isses when it occurs in communication spacecrafts or Weather or GPS satellites.
- Are there seasonality in system behavior? Is it diffent during night and day time. In case of spacecrafts, the battery behaviors and temperature change rapidly. Spacecrafts can show various dynamics as seasons change (summer, winter, fall, etc) depending upon its exposure to sun which is the primary means of charging batteries.
- How reduce False positives? In case of spacecraft operations, there are operations such as **Station keeping**, **North South** or **East West** station keeping operations (AKA maneuvers) that are done to align the spacecraft pointing to antennas to enable **Communication Sub System (CSS)** to operate normally. These maneuvers are difficult to capture in general dynamics - how shouls the system handle these normal and yet difficult to capture dynamics of the system.
- How should data be preprocessed for various algorithsms? For LSTM, it is critical to normalize the data; for SIAT, critical to eliminate categorical variables. In all cases, the data must be numeric and quality tested.
- How to capture the **"false positive"** patterns and to apply post-processing to reduce the false alarms.
- How to customize **automated actions** upon known anomalous patterns.
- How not to miss **True Positives**
- How to conduct feature engineering to detemine most critical sensors
- How to trigger anomaly if if it occurs in sub-space (this is especially true in case higher dimentional space of 100k sensors where anomaly may be caused due to small deviations in sensors)
- How to evolve the model; Use this to show how the system is aging and offer insights to robust design and operations.

- What is the **Concept of Operations** (CONOPS). When to build the model and how to deploy. The computation power requirements for developing the model and power requirements for inferences.
- Where does the computation run. This is especially important when the anomaly detection framework needs to be deployed in Space environments. In addition, power, memory requirements plays a vital role in deplying system where it is expected to discover abd self-calibrate.
- How to handle diffent types of data? Most algorithms are not capable of handling categorical or binary type of data. (For example, switches ON/OFF or status reporting switches). How to holistically handle all types of data.
- How to augment (or enrich) the data set and remove irrelvant data. (For ex. a sensor that is high correlated with time is rather not have much signals and thus can be omitted)
- How to handle "Log data", text data or Image data that have charateristics of time series.
- How to develop of user interface for vairous stake holders.
- Determine the sufficient data quantity to capture the system dynamics.
- How to patch or handle missing data. This is especailly true in spacecraft (or any high dimentional dataset) there will be missing or bad-quality data (anomaly); how to detect and patch before capturing the model to reduce false positives.

> Data patching technics such as "forward fill", backward-fill are not necesarily effective always.

- How to predict future anomalies - i.e. capture the trending signals that leads to anamoly
- How to use it for maintenance

Type *Markdown* and LaTeX: α^2

In [ ]:

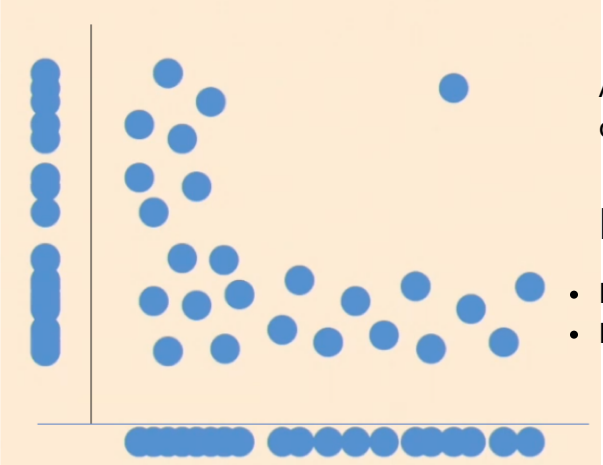
```
%run 00_basic.ipynb
```

Anomaly Detection using Isolation Forest

An **Isolation Forest** is to build trees recursively. In each iteration:

- Pick a random feature
- Split the data based randomly chosen value b/w the minimum and maximum of the chosen feature.
- Repeat until the entire data set is partitioned to form an individual tree in the forest.

Anomalies generally have shorter paths from the root than normal. An anomaly score to a point is the average path length from root to it.



A sample set of points is shown and if we see the points in one dimension, one can see that there are no anomalies.

Draw backs

- Does not work with categorical variables inherantly
- Not a good one for missing values

In [ ]:

```

```

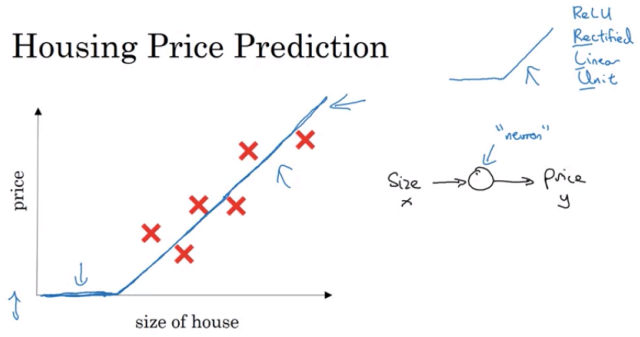
In [ ]:

```

```

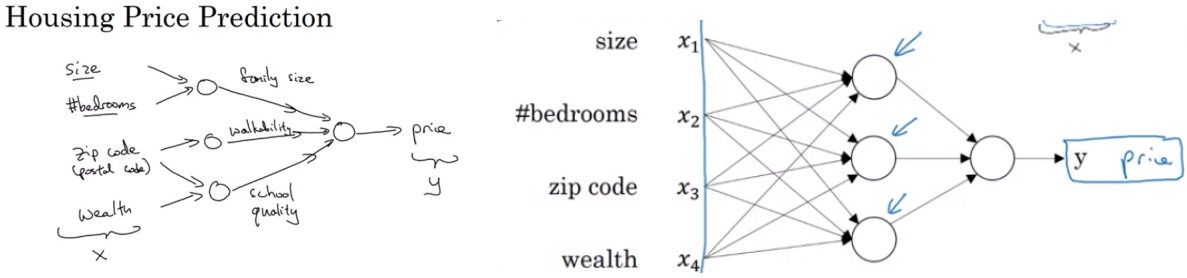
1. Neural Networks Introduction

Consider a simple function of predicting the house prices based on the size of the house. This is a simple function that finds a linear fit of the data and also a simple neural network that takes a size of the house and outputs the price.



A simple network fits a function to points shown in cross marks. The resulting function is called RELU function (Rectified Linear Unit) which will be described later. However, the methods to fit the function to set of predictors is what NN does. The term **deep learning** refers to training deep neural networks which will be discussed later.

If we have multiple features such as, size, #bedrooms, Zip code etc. then we may have lot more cells (or neurons).



A Neural Network is a set of nodes that fits a function to predict output.

1.1. Notations

m	Size of length of training examples	n	Length of predictors $X = \|X\|$
X	Notation for input predictors		Super script (i) represents the ith example

2. Binary classification

Binary classification is given an input, classify the data into one or other class. As an example: if we are given a set of images, classify them as pictures of cat or non-cats.

2.1. Logistic regression

Logistic regression is used for classification problems. The term **regression** here seems to indicate otherwise. Model predicts a number between 0 and 1 and a threshold is used to compare the result to predict one or other depending upon if it is greater or lesser than the threshold. A threshold T (usually 0.5) which indicates if it the predicted number is less than threshold T then we predict one otherwise other class.

Here we are restricting for two classes - however, later we show how it can be extended to show multiple classes.

In Linear regression we had the hypothesis function

$$h = \theta^T X \tag{1}$$

We would like to interpret h as a probability function. As is, it is not a function that can do binary classification.
Therefore if we consider the **logit** AKA **log of odds** where P is the probability. And X is the input:

$$P = g(x)$$
$$\frac{P}{1-P} = z = \theta^T X$$

g is a function that translates h *to* P the probability give X

Given X we want to compute the output $\hat{y}$ = P(y = 1 | X) ==> Probability that y = 1 give X

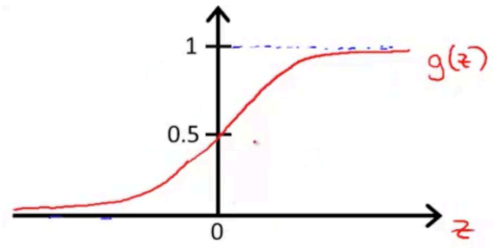
Solving for P, we get:

$$P = \frac{1}{1 + e^{-z}}$$

Therefore, in Logistic Regression, the form of the hypothesis takes the form:

$$h(x) = g(\theta^T x)$$
$$g(\theta^T x) = \frac{1}{1+e^{-z}}$$
$$z = \theta^T x$$

g is some function that transforms linear equation to probability



The function g is called Logistic function or Sigmoid function. Logistic or Sigmoid can be used interchangeably.

It is shown on the left. When $z >> 0$ the function reaches 1, when $z << 0$, it reaches zero, when at 0, $g(x)$ is 0.5. when z is +ve the term $e^{-z} < 1$, therefore, $\frac{1}{1+e^{-z}}$ is $\frac{1}{<2} > .5$; and when z is -ve the denominator is > 2 and thus g will be < 0.5

$$g = \begin{cases} \frac{1}{1+e^0} = 0.5, & \text{if } z = 0 \\ > .5, & z < 1 \\ < 0.5, & z > 1 \end{cases}$$

In Linear regressions we had the cost function

$$Cost(\text{AKA Loss function}\mathscr{L}) = J(\theta) = \frac{1}{2m}\sum_{1}^{m}(\hat{y} - y)^2$$ sum of squares of residues

We need to modify this because $(\hat{y} - y)^2$ is non-convex function for y *in* $\{0, 1\}$. If you plot it it will look wavy with no one max or minima. Therefore, we want a convex function so that we can gradient decent converge to global minimum.

In Logistic Regression, given a X examples :

$(x^{(1)}, y^1), (x^{(2)}, y^2), \ldots (x^{(m)}, y^m)$, we want to compute $\hat{y} \approx y^{(i)}$

Where $\hat{y} = \frac{1}{1+e^{-z}}$ at least in the training example want it to be as close match as possible.

We define the cost function for single training example:

$$Cost(\hat{y}, y) = \begin{cases} -\log \hat{y}, & \text{if } y = 1 \\ -\log(1-\hat{y}) & \text{if } y = 0 \end{cases}$$

If y is 1, then want $\hat{y}$ to be close to 1 anything less is incurred as cost

If y is 0, then want $\hat{y}$ to be close to 0, otherwise cost is the difference.

Since log of number in [0, 1] is negative, we take the negative log. The above equation can be combined:

$[-y_i \log \hat{y} - (1 - y_i)\log(1 - \hat{y})]$ when y_i is 0 first term goes away and similarly other term

The cost function for the entire training set is:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)) \qquad (2)$$

Now all that remains to be done is to find θ to minimize $J(\theta)$ that is convex so that the slope is zero (first derivate) is zero.

We now have:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)) \qquad \text{Same as (2) above}$$

NOTE - We want to find θ that minimize the cost function. In order to find the minimum,

2.1. We take the first order differentiation of the equation to be minimized
2.2. Find the roots from step (1) - Since Cost function is convex, there should be only one root.

That is what we will do next,

where $h_\theta(x)$is defined as follows:

$$\hat{y} = h_\theta(x) = g(\theta^T x)$$
$$g(z) = \frac{1}{1 + e^{-z}} \qquad (4)$$
$$\frac{\partial}{\partial \theta_j} J(\theta) = \sum_{i=1}^{m} (\hat{y} - y^i) x_j^{(i)}$$

Detailed Derivation: Now, we show detailed derivation of equation (4).

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1-y^i)\log(1-\hat{y}^i))$$
$$\theta x^i = \theta_0 + \theta_1 x_1^i + \ldots + \theta_p x_p^i .$$

Lets just consider and notice,

$$\log \hat{y}^i = \log \frac{1}{1+e^{-\theta x^i}} = -\log(1+e^{-\theta x^i})$$

$$\log(1-h_\theta(x^i)) = \log(1-\frac{1}{1+e^{-\theta x^i}}) = \log(e^{-\theta x^i}) - \log(1+e^{-\theta x^i}) = -\theta x^i - \log(1+e^{-\theta x^i}),$$

[this used: $1 = \frac{(1+e^{-\theta x^i})}{(1+e^{-\theta x^i})}$, the 1's in numerator cancel, we used: $\log(x/y) = \log(x) - \log(y)$]

Since our original cost function is the form of:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(h_\theta(x^i)) + (1-y^i)\log(1-h_\theta(x^i))$$

Plugging in the two simplified expressions above, we obtain

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[-y^i(\log(1+e^{-\theta x^i})) + (1-y^i)(-\theta x^i - \log(1+e^{-\theta x^i}))\right]$$

which can be simplified to:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y_i\theta x^i - \theta x^i - \log(1+e^{-\theta x^i})\right] = -\frac{1}{m}\sum_{i=1}^{m}\left[y_i\theta x^i - \log(1+e^{\theta x^i})\right], \quad (*)$$

where the second equality follows from

$$-\theta x^i - \log(1+e^{-\theta x^i}) = -\left[\log e^{\theta x^i} + \log(1+e^{-\theta x^i})\right] = -\log(1+e^{\theta x^i}) .$$

[we used $\log(x) + \log(y) = log(xy)$]

All you need now is to compute the partial derivatives of (*)w.r.t. θ_j.
As

$$\frac{\partial}{\partial\theta_j}J(\theta) = \frac{\partial}{\partial\theta_j}\Big(-\frac{1}{m}\sum_{i=1}^{m}\Big[y_i\theta x^i - \log(1+e^{\theta x^i})\Big]\Big)$$

$$\frac{\partial}{\partial\theta_j}y_i\theta x^i = y_i x_j^i,$$

$$\frac{\partial}{\partial\theta_j}\log(1+e^{\theta x^i}) = \frac{x_j^i e^{\theta x^i}}{1+e^{\theta x^i}} = x_j^i h_\theta(x^i)$$

Thus,

$$\frac{\partial}{\partial\theta_j}J(\theta) = \sum_{i=1}^{m}(h_\theta(x^i) - y^i)x_j^i \qquad \blacksquare$$

Once we know the derivative of the *Cost* function. Next step is to use it to find the parameters, θ that minimizes the cost function.

2.3. Gradient Decent (GD)

To recap, we have

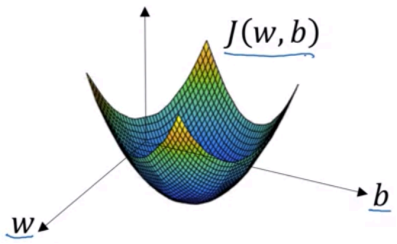
$$\hat{y} = \sigma(\theta^T X + b) \qquad \text{we did not speak about b before}$$

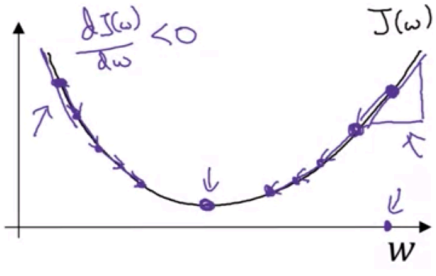
$$\sigma(z) = \frac{1}{1+e^{-z}} \qquad 2.2.1$$

$$J(\theta, b) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1-y^i)\log(1-\hat{y}^i)) \qquad 2.2.2$$

$$\frac{\partial}{\partial\theta_j}J(\theta) = \sum_{i=1}^{m}(\hat{y}^{(i)} - y^{(i)})x_j^i \qquad 2.2.3$$

We can use the above equations in *Gradient Descent* algorithm to find the optimal (θ, *b)* to minimize the cost function *J(θ)*. The rationale is as follows. Any convex function would have a graph as shown below , if we plot :





Procedure

1. Initialize (θ, *b)* to any random value
2. Compute the derivative as given in 2.2.3
3. Update θ as $\theta = \theta - \alpha \cdot \frac{\partial}{\partial \theta_j} J(\theta)$ where alpha is learning rate determines how big of a step we take each time.
4. Continue until the error reaches acceptable limit.

Why does this work. Let's take an example as shown in the right side of the figure above. If the point is on the right side, the slope (or the derivative is +ve) therefore we subtract from θ. On the other hand, if it is on the left of the minimum the slope is -ve and we move θ towards right. Therefore GD will move θ in the right direction to find minimum point.

2.4. Rationale for Logistic Regression loss function.

$\hat{y} = \sigma(\theta^T X + b)$

We want to interpret:

$$\hat{y} = P(Y = 1 \mid X) : \begin{cases} \text{if } \; y = 1 \; \text{ then } \; P(y = 1 \mid X) \; = \; \hat{y} \\ \text{if } \; y = 0 \; \text{ then } \; P(y = 0 \mid X) \; = \; 1 - \hat{y} \end{cases}$$

We can write both in one equation:

$$P(Y \mid X) = \hat{y}^{y}(1 - \hat{y})^{(1-y)} \qquad 2.3.1$$

when y is 0, $P(Y/X) = (1 - \hat{y})$; similarly other case

We want to maximize this across all examples: $P = \prod_{i=1}^{m} P(y \mid X)$

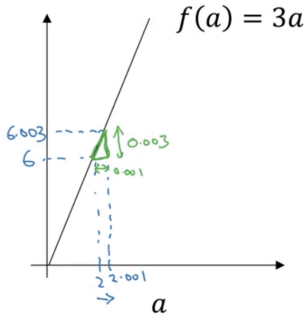
Maximizing the probability is equivalent to maximizing the log of the probability. Therefore,

$$log(P_{\text{all training examples}}) = log\left(\prod_{i=1}^{m} P(y^i \mid X^i)\right) = \sum_{i=1}^{m} log(P(y^i \mid X^i)) \qquad 2.3.2$$

From 2.3.1, $log(P(y \mid X) = y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i))$

We want to maximize P and minimize the loss that is -ve of P, therefore, we want to choose parameters that minimizes the cost of the function that can be does using maximum likelihood estimation methods. Therefore the final cost function from 2.3.2 averaged over all training examples is:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i))$$



■

3. Derivatives

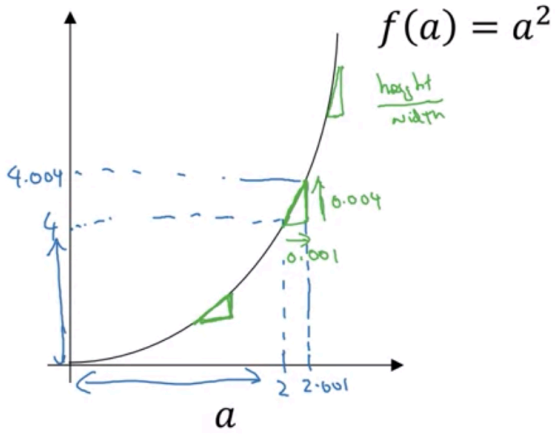
3.1. Intuitive definition of derivative

Suppose if we have a function:
f(a) = 3a when plotted, it looks like in the figure as shown.
Derivative is just a slope of the e=function evaluated at any point *x*.

For example at *x = 2*, slope is height divided by width. Suppose if we take small width 0.001

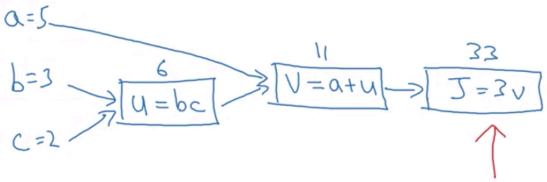
(6.0001 - 6.0003) / (2 - 2.0001) = 3.
For this function, slope is 3 wherever you take *x*.

In another example, where slope is different at different points.
It is shown in the example in the picture. At point *a = 2,* slope is 4, when *a* = 5, the slope is 10.
I n



$a = 2$ $\quad f(a) = 4$
$a = 2.001$ $\quad f(a) \approx 4.004$ (4.004001)
slope (derivation) of $f(a)$ at $a = 2$ is 4.
$\frac{d}{da} f(a) = 4$ when $a = 2$.
$a = 5$ $\quad f(a) = 25$
$a = 5.001$ $\quad f(a) \approx 25.010$
$\frac{d}{da} f(a) = 10$ when $a = 5$

general for this slope or the derivative is *2a.*



As you can see, we took a small nudging at x of 0.001, but we want this to be infinitely small. In general:

$$\lim_{\Delta a \to 0} \frac{f(a) - f(a + \Delta a)}{a + \Delta a - a} = \lim_{\Delta a \to 0} \frac{f(a) - f(a + \Delta a)}{\Delta a}$$

3.2. Computation graphs

In order to compute the derivatives of the cost function, it gets complicated; therefore computation graphs helps to break down the complex functions to make it easier to compute derivates. It also helps to create automated derivation code. Let see an example.

Suppose we have a function *J(a,b,c) = 3 (a + b c)*, we want to find the *J'* - derivative of *J w.r.t. a,b,c*.

Lets write this as:

$$u = bc$$
$$v = a + v$$
$$J = 3v$$

We can write this as graph as shown here.
To find derivative of *J* w.r.t. *a, b, c,* first find the derivative of *J w.r.t v,* which is *3*.
Next, we find *J'* w.r.t *u,* we can do this as follows:

$$\frac{dJ}{du} = \frac{dJ}{dv} \cdot \frac{dv}{du} \qquad \text{we already know } \frac{dJ}{dv} = 3; \text{ and } \frac{dv}{du} = 1 \Rightarrow \frac{dJ}{du} = 3$$

Similarly, $\frac{dJ}{da} = \frac{dJ}{dv} \cdot \frac{dv}{da} = 3 \cdot 1 = 3 = 3$

Continuing this way,

$$\frac{dJ}{db} = \frac{dJ}{du} \cdot \frac{du}{db} = 3 \cdot c$$
$$\frac{dJ}{dc} = \frac{dJ}{du} \cdot \frac{du}{dc} = 3 \cdot b$$

3.3. Logistic regression - Back propagation

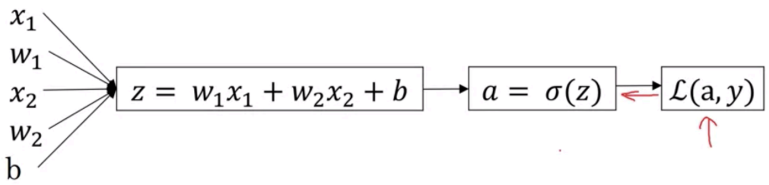
Lets consider a Logistic regression with only two predictors

$$a = \sigma(WX + b)$$
$$z = WX + b = w_1 . x_1 + w_2 x_2 + b$$
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Loss function $\mathscr{L}(a, y) = -(y \log(a) + (1 - y)\log(1 - a))$

These are exactly same as before except we replace y-hat with a that stands for activation function. (also makes typing so much easy)

Computation graph is



To back propagate we need to know how to adjust w, w2 and b to minimize the loss. This means, we need to find derivative of Loss function w.r.t to w1, w2, b and use it adjust the weights to get to the minimum.

From the computation graph,

$$\frac{d\mathscr{L}}{dw_1} = \frac{d\mathscr{L}}{da}\frac{da}{dz}\frac{dz}{dw_1} \tag{I}$$

$$\frac{d\mathscr{L}}{da} = \frac{-y}{a} + \frac{1-y}{1-a} \qquad \text{because: } log(1-a) = \frac{-1}{1-a}$$

$$= \frac{-(1-a)y + a(1-y)}{a(1-a)} = \frac{a-y}{a(1-a)} \tag{A}$$

next Consider:

$$\frac{da}{dz} = \frac{(1+e^{-z})\cdot 0 + e^{-z}}{(1+e^{-z})^2} \qquad \text{reminder:} a = \frac{1}{(1+e^{-z})}$$

$$= \frac{e^{-z}}{(1+e^{-z})^2} \tag{B}$$

Notice:

$$a = \frac{1}{1+e^{-z}} \therefore (1-a) = \frac{1+e^{-a}-1}{1+e^{-z}} = \frac{e^{-z}}{1+e^{-z}} \tag{C}$$

.

Substituting (C) in (B):

.

$$\frac{da}{dz} = a(1-a) \tag{D}$$

.

Finally:

.

$$\frac{dz}{dw_1} = x_1 \tag{E}$$

.

Sbstituing (A), (D), (E) in (I):

.

$$\frac{d\mathscr{L}}{dw_1} = \frac{d\mathscr{L}}{da}\frac{da}{dz}\frac{dz}{dw_1}$$

$$= \frac{(a-y)}{a(1-a)} a(1-a)\cdot x_1$$

$$\frac{d\mathscr{L}}{dw_1} = (a-y)x_1$$

Similarly we get:

$$\frac{d\mathscr{L}}{dw_2} = (a-y)x_2$$

$$\frac{d\mathscr{L}}{db} = (a-y)$$

In general the derivative of loss function will be multiplying corresponding *(a-y)* with the predicator.

With this we can implement back-propagation as follows:
For one training example compute the following to adjust weights:

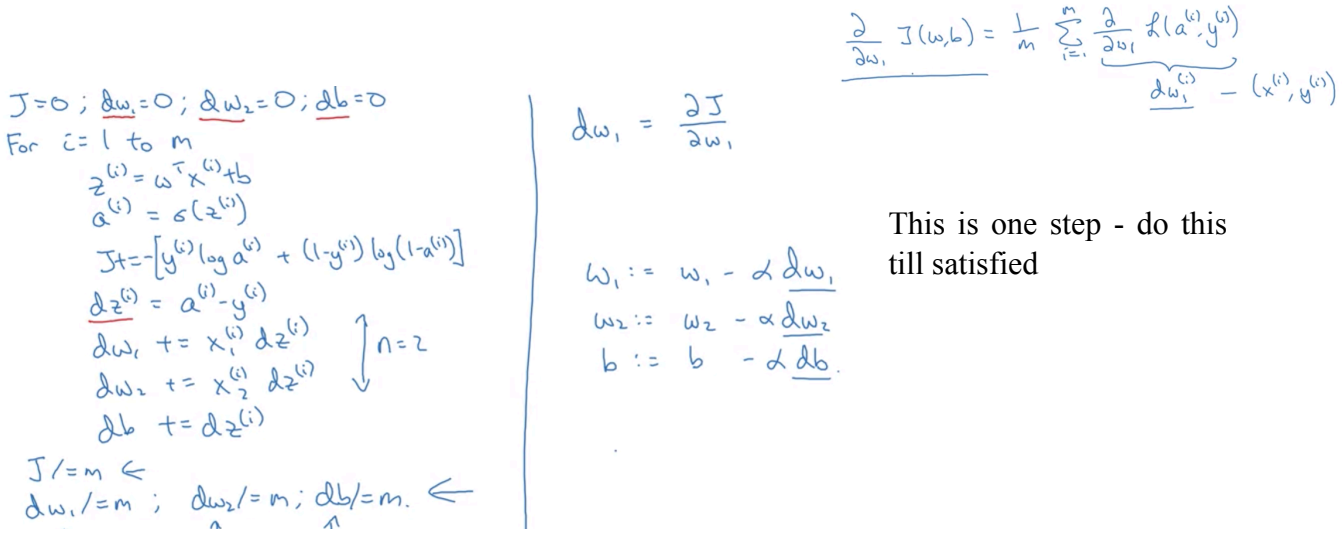
```
dz    =     (a - y)
dw1   =     dz * w1
dw2   =     dz * w2
db    =     dz

# Update
w1    = w1 - alpha * dw1
w2    = w2 - alpha * dw2
b     = b - dz
```

Repeat until error is minimized; Not a great algorithm. A complete example for m training example is shown below.

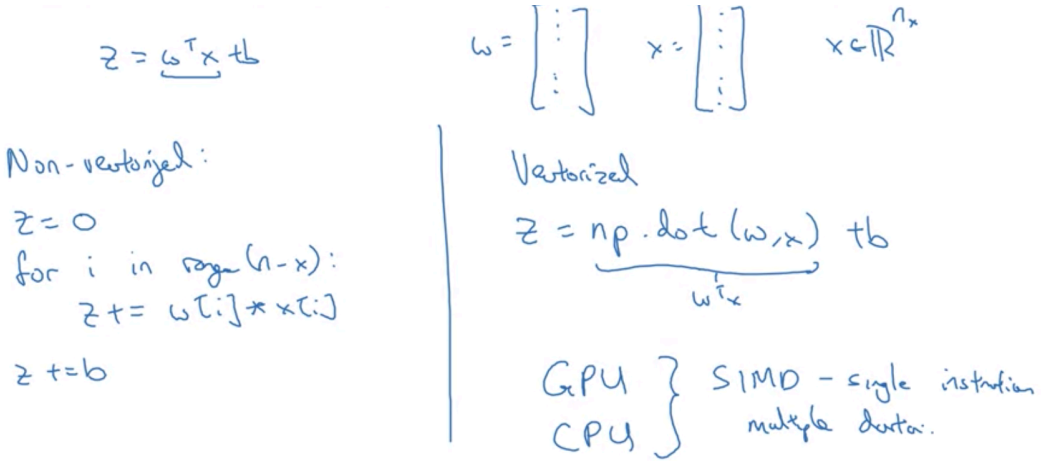
3.4. Gradient Descent with loss function to adjust weights.

As show before the loss function for one training example. It so happens that for *m* training examples, we can use it compute the total cost function.



3.5. Vectorization.

A mechanism to get rid of explicit for-loops.

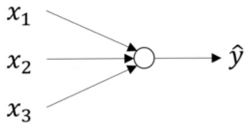


3.6. Notes.

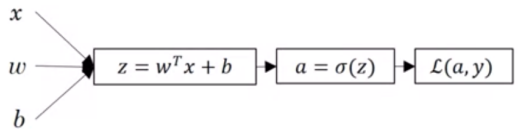
In python, arrays shape look like (5,) - they are called rank 1 arrays.

A = np.random(5) creates a vector a.shape = (5,) => avoid using the rank 1 shaped vectors.

4. Neural Networks

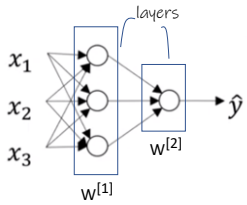


A logistic regression (also a neural network with one node) looks like the figure.

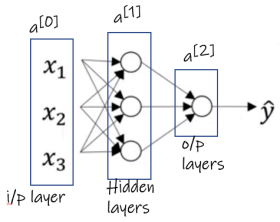


The inputs are passed in and through back propagation, W is adjusted and refined to find the optimal values for W and b to minimize the cost function.

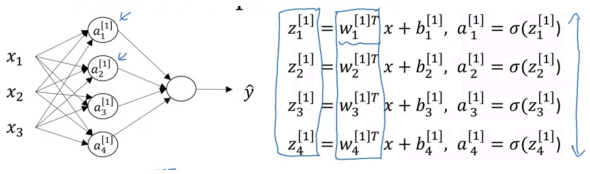
A Neural Network may consists of more than one such node stacked, also possibly arranged in layers to create a complex network. Each layer will compute z and a and finally last layer will compute the loss function $\mathscr{L}$.



A various types of Neural Network. A two layer network is shown. Two because, most times, input layer is not counted as a layer.



4.1. Computations of weights in NN.



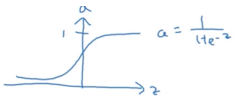
Similar to logistic regression in forward propagation, for each cell we compute the activations Z1, Z2 etc. Think each cell as one logistic regression cell. The back propagation happens for each cells very similar to one cell logistic regression and it happens to every cell to back propagate.

4.2. Activation Functions.

Sigmoid, tanh, RELU, Leaky RELU

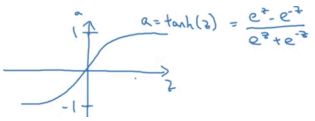
4.2.1. Sigmoid functions

Sigmoid function that is shown that we have been using so far



4.2.2. tanh functions

tanh function

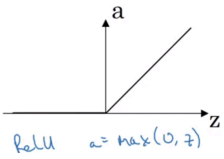


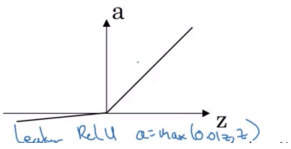
$g'(z) = \frac{d}{dz} g(z)$ = slope of $g(z)$ at z

$= 1 - (\tanh(z))^2$ ←

4.2.3. RELU functions

Or Leaky RELU function, shown below





$g'(z) = \begin{cases} 0 & \text{if } z < 0 \\ 1 & \text{if } z \geq 0 \end{cases}$

~~undef if $z = 0$~~

$z = 0.0000\ldots0$

4.3. Take aways.

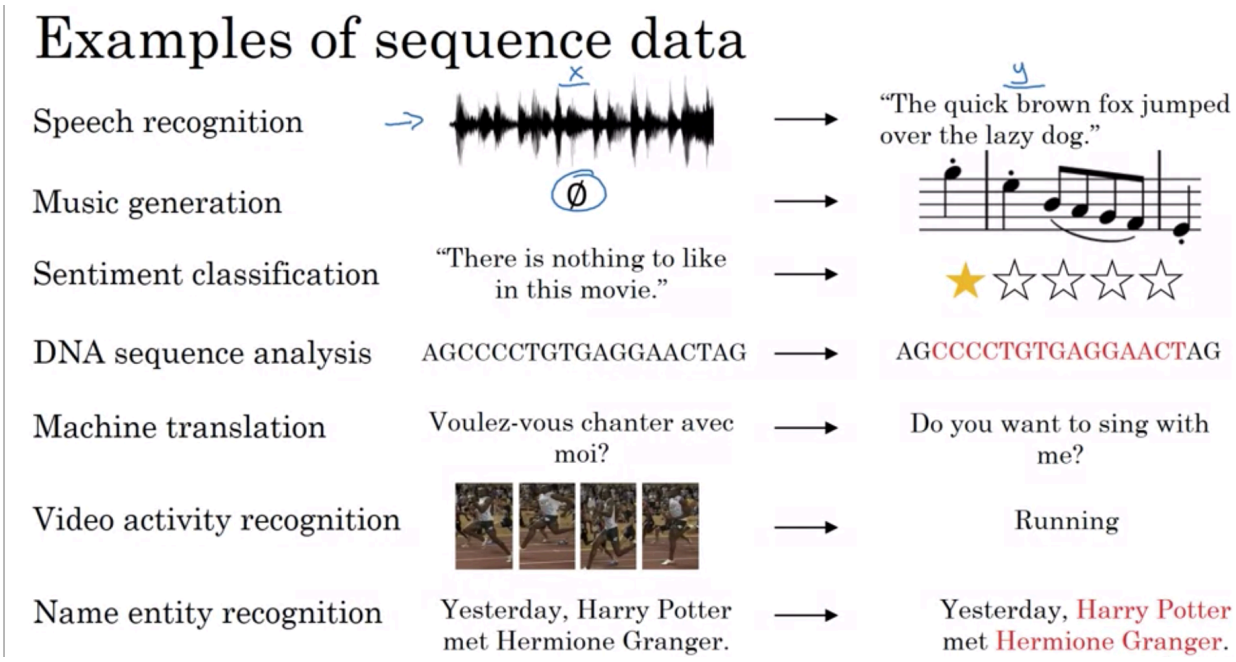
4.3.1. Never use sigmoid in hidden layers except output layers
4.3.2. If you are not sure what to use as activation for hidden layer, choose RELU
4.3.3. Do not use Linear Activation function in hidden layer - it practically makes layers useless.
4.3.4. Only place where linear activation function may be used is in output layer

5. Implementation

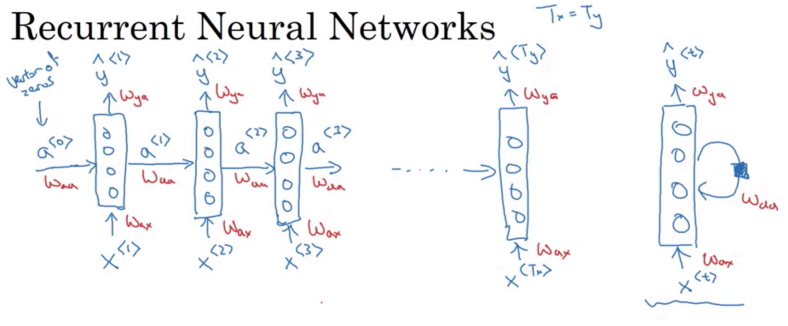
Keras implementation are in 01_NN-Keras.ipynb

6. Sequence Models

6.1. Examples



6.1. Notations

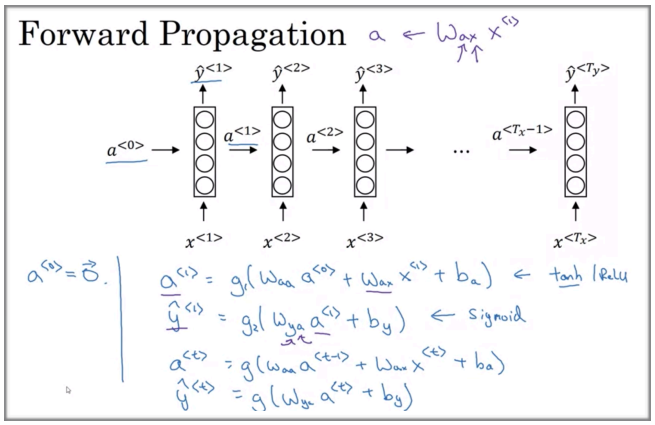


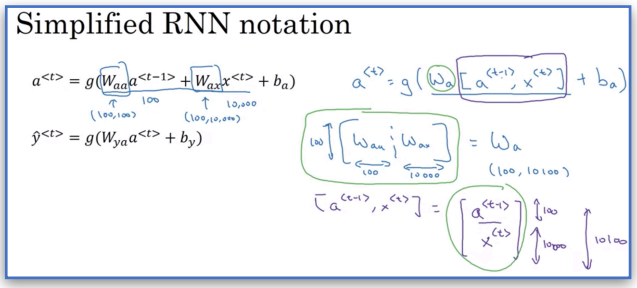
$x^{(i)<t>}$: input at time t T_x : Length of input sequence

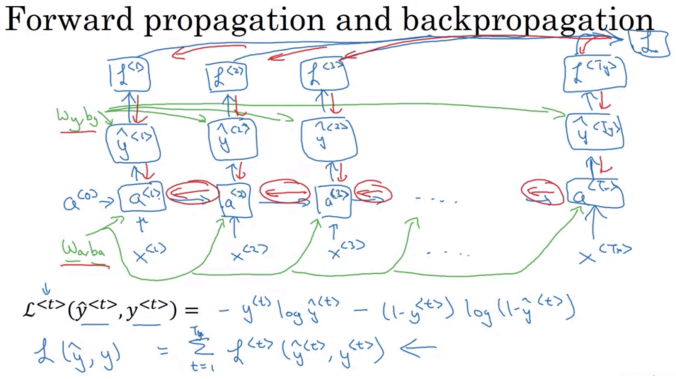
$y^{(i)<t>}$: input at time t T_y : Length of input sequence

Also note that we have bi-directional RNN

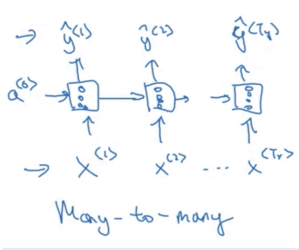
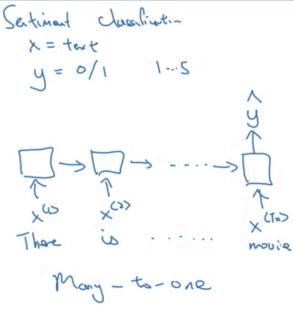
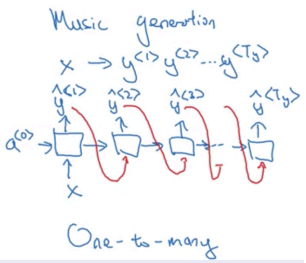
6.1. Calculations







6.1. Types of RNN

Many to Many	Many to 1	1 to many	1 to 1
			

Tx, Ty can be different Ex: translation	Sentiment classification	Music Generation	Becomes generic simple NN

6.1. Language modeling

Given a set of sentence, which sentence is more likely

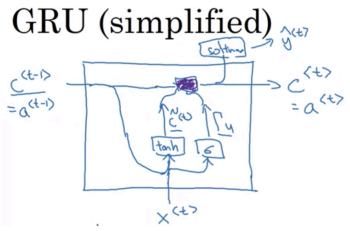
The apple and pair salad ex: P = 3.2
The apple and pear salad P = 5.7 <== this is more likely

6.1. Vanishing/exploding gradient

In a much deeper NN, then the gradient is difficult to propagate up. The errors in later step to affect the weights in earlier in the layer. Therefore outputs are influenced strongly influenced by local nodes.

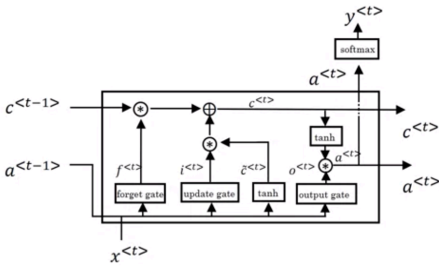
Vanishing gradients can cause the weights to almost becomes zero. Exploding gradients is where weights increase astronomically. Exploding gradients are easy to spot and easy to manage by gradient clipping. Vanishing does more things to do to handle.

6.1. Gated Recurrent Unit (GRU)



$$\tilde{c}^{<t>} = \tanh(W_c[\Gamma_r * c^{<t-1>}, x^{<t>}] + b_c)$$
$$\Gamma_u = \sigma(W_u[\, c^{<t-1>}, x^{<t>}] + b_u)$$
$$\Gamma_r = \sigma(W_r[\, c^{<t-1>}, x^{<t>}] + b_r)$$
$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + (1 - \Gamma_u) * c^{<t-1>}$$
$$a^{<t>} = c^{<t>}$$

$$\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$$
$$\Gamma_u = \sigma(W_u[\, a^{<t-1>}, x^{<t>}] + b_u)$$
$$\Gamma_f = \sigma(W_f[\, a^{<t-1>}, x^{<t>}] + b_f)$$
$$\Gamma_o = \sigma(W_o[\, a^{<t-1>}, x^{<t>}] + b_o)$$
$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$
$$a^{<t>} = \Gamma_o * \tanh c^{<t>}$$



6.1. Long Short Term memory (LSTM)