Hypothesis Testing

Hypothesis Testing refers to assumptions. We are testing our assumptions.
To testing the assumptions, we need statistical testing tools.

It is all about finding whether the said assumptions/ the said statistical concept is present or absent in the data.

statistical concept:

1- **Test for Normalization**
2- **Test for correlation**
3- **Test for feature elimination**

There are two types of **Hypothesis Testing**:

1- NULL Hypothesis (Negative answer to your question)
2- Alternate Hypothesis (Positive answer to your question)

When it comes to performing any statistical test on a dataset, you need to perform 5 steps:

Step 1: Create a viable question(The question should be yes or no question)
Step 2: convert the question into Hypothesis
NULL ➔
ALTERNATE ➔
Step3: Select the statistical test and formula to perform.
Step4: Select the SL (0.05, 0.01, 0.1, data scientist)
Step5: Find the P-value and compare with SL to identify who wins.

Imagine we want to test:

1- Whether Sugar is sweet or not? ➔ (Created a viable question)
2- Sugar is **NOT** sweet. ➔ (convert the question into **NULL Hypothesis**)
Sugar is sweet. ➔ (convert the question into **Alternate Hypothesis**)
3-

Statistical Tests:

1- Correlation Test
2- Normality Test
3- Non-parametric Test
4- Parametric Test
5- Chi-square Test

Correlation Test, Normality Test, Non-parametric Test, Parametric Test ➔ **applied on Quantitative Data(Numerical Data)**
Chi-square Test ➔ **Qualitative Data(Categorical Data)**

Correlation Test: ➔ goal of correlation test is identify correlation between two variables**(feature and label)**

1- Pearson's Correlation Test
2- Spearman Rank Test
3- Kendall Tau Rank Test

Normality Test: ➔ the primarily goal of Normality test is identify whether the given column follows normal distribution or not? **(just one feature)**

1- Shapiro Test
2- Anderson Darling Test
3- Normal Test

Non-parametric Test ➔ goal of Non-parametric Test whether two features have any kind of relationship. If The two features **fail** normality test and then go for Non-Parametric Test **(feature and feature) (Use for feature elimination):**

1- Wilcoxon Test
2- Matt – Whitney U test
3- Kruskal-Wallis(H) Test
4- Friedman Test

Parametric Test ➔ goal of Non-parametric Test whether two features have any kind of relationship. There is a catch here, you should use **Parametric Test** if normality passes. The two features **pass** normality test and then go for Parametric Test **(feature and feature) (Use for feature elimination)**

1- Student t-Test
2- Paired Student t-Test
3- ANOVA

Chi-square Test:

1- Chi-square Test

Practical Test

1		`data = pd.read_csv('50_Startups.csv')`

2		`data.info()`
		
		`<class 'pandas.core.frame.DataFrame'>`
		`RangeIndex: 50 entries, 0 to 49`
		`Data columns (total 5 columns):`
		` #  Column  Non-Null Count  Dtype`
		`---  ------  --------------  -----`
		` 0  R&D Spend  50 non-null  float64`
		` 1  Administration  50 non-null  float64`
		` 2  Marketing Spend  50 non-null  float64`
		` 3  State  50 non-null  object`
		` 4  Profit  50 non-null  float64`
		`dtypes: float64(4), object(1)`
		`memory usage: 2.1+ KB`

3	**Correlation Test**
	1. Pearson's Correlation Coeff Test
	(R&D Spend and Profit)
	
	Step1: Create A Viable Question
	Question: **Lets Test whether R&D Spend and Profit have a Linear Relationship?**
	
	Step2: Convert the Question into Hypothesis Types
	
	Null Hypothesis : R&D Spend and Profit have NO Linear Relationship
	Alternate Hypothesis: R&D Spend and Profit have Linear Relationship
	
	Step3: Select The Statistical Test to Perform: ---> **Pearson's Correlation Test**
	
	Scipy --> Scientific Python -- All formulaes related to Math and Stat are present in Scipy
	
	Step4: Select the **Significance Level (SL = 0.05)**
	
	Step5: Find **the p-value of R&D Spe**nd using Pearson's Correlation Test
		SL = 0.05
		
		from scipy.stats import pearsonr
		
		corr, pvalue = pearsonr(data['R&D Spend'], data['Profit'])
		
		if pvalue <= SL:
		  print("Alternate Hypothesis passed (H1) -- R&D Spend and Profit have Linear Relationship")
		else:
		  print("Null Hypothesis passed (H0) -- R&D Spend and Profit have NO Linear Relationship")
		
		Alternate Hypothesis passed (H1) -- R&D Spend and Profit have Linear Relationship

	1.Pearson's Correlation Coeff Test
	(Marketing and Profit)
	
	Step1: Create A Viable Question
	Question: Lets Test whether **Marketing and Profit** hava a Linear Relationship?
	
	Step2: Convert the Question into Hypothesis Types
	Null Hypothesis: Marketing and Profit have NO Linear Relationship
	Alternate Hypothesis: Marketing and Profit have Linear Relationship
	
	Step3: Select the Statistical Test to Perform: --->
	Pearson's Correlation Test
	
	Step4: Select the Significance Level **(SL = 0.05)**
		SL = 0.05
		
		from scipy.stats import pearsonr
		
		corr, pvalue = pearsonr(data['Marketing Spend'], data['Profit'])
		
		if pvalue <= SL:
		  print("Alternate Hypothesis passed (H1) -- Marketing and Profit have Linear Relationship")
		else:
		  print("Null Hypothesis passed (H0) -- Marketing and Profit have NO Linear Relationship")
		
		Alternate Hypothesis passed (H1) -- Marketing and Profit have Linear Relationship

	Step5: Find the **p-value of Marketing** using Pearson's Correlation Test	

	1.Pearson's Correlation Coeff Test (Administration and Profit)
	
	Step1: Create A Viable Question
	Question: Lets Test whether Administration and Profit hava a Linear Relationship?
	
	Step2: Convert the Question into Hypothesis Types
	
	Null Hypothesis : Administration and Profit have NO Linear Relationship
	Alternate Hypothesis: Administration and Profit have Linear Relationship
	
	Step3: Select The Statistical Test to Perform: ---> Pearson's Correlation Test
	
	Step4: Select the Significance Level (SL = 0.05)
	
	Step5: Find the p-value of Administration using Pearson's Correlation Test
		`SL = 0.05`
		
		from scipy.stats import pearsonr
		
		corr, pvalue = pearsonr(data['Administration'], data['Profit'])
		
		if pvalue <= SL:
		  print("Alternate Hypothesis passed (H1) -- Administration and Profit have Linear Relationship")
		else:
		  print("Null Hypothesis passed (H0) -- Administration and Profit have NO Linear Relationship")
		
		Null Hypothesis passed (H0) -- Administration and Profit have NO Linear Relationship

2. Spearman Rank Test (correlation test)

Test whether **R&D Spend** and **Profit** have a Linear Relationship?

	Step1: Create A Viable Question
	Question: Lets Test whether **R&D Spend** and **Profit** have a Linear Relationship?
	
	Step2: Convert the Question into Hypothesis Types
	
	Null Hypothesis: R&D Spend and Profit have NO Linear Relationship
	Alternate Hypothesis: R&D Spend and Profit have Linear Relationship
	
	Step3: Select the Statistical Test to Perform: ---> **Spearman RankTest**
	
	Step4: Select the Significance Level (SL = 0.05)
	
	Step5: Find the p-value of R&D Spend using Spearman Rank Test
		SL = 0.05
		
		from scipy.stats import spearmanr
		
		corr, pvalue = spearmanr(data['Administration'], data['Profit'])
		
		if pvalue <= SL:
		print("Alternate Hypothesis passed (H1) -- Administration and Profit have Linear Relationship")
		else:
		print("Null Hypothesis passed (H0) -- Administration and Profit have NO Linear Relationship")
		
		Null Hypothesis passed (H0) -- Administration and Profit have NO Linear Relationship

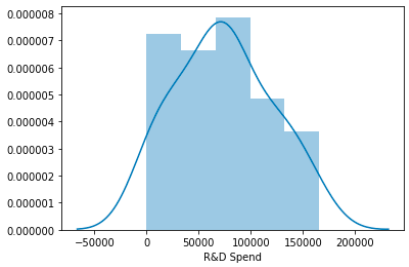
3.Kendall tau Test(Correlation Test)

	Step1: Create A Viable Question
	Question: Lets Test whether **R&D Spend and Profit** hava a Linear Relationship?
	
	Step2: Convert the Question into Hypothesis Types
	
	Null Hypothesis : R&D Spend and Profit have NO Linear Relationship
	Alternate Hypothesis: R&D Spend and Profit have Linear Relationship
	
	Step3: Select The Statistical Test to Perform: ---> **Kendall Test**
	
	Step4: Select the Significance Level **(SL = 0.05)**
	
	Step5: Find the **p-value of R&D Spend using Kendall Test**
		SL = 0.05
		
		from scipy.stats import kendalltau
		
		corr, pvalue = kendalltau(data['Administration'], data['Profit'])
		
		if pvalue <= SL:
		print("Alternate Hypothesis passed (H1) -- Administration and Profit have Linear Relationship")
		else:
		print("Null Hypothesis passed (H0) -- Administration and Profit have NO Linear Relationship")
		
		Null Hypothesis passed (H0) -- Administration and Profit have NO Linear Relationship

Normality Test:

to check whether the given column is normally distributed or not?

	Shapiro
	
	Step1: Create A Viable Question
	Question: **Lets Test whether R&D Spend is normally distributed?**
	
	Step2: Convert the Question into Hypothesis Types
	
	Null Hypothesis : R&D Spend is NOT normally distributed
	Alternate Hypothesis: R&D Spend is normally distributed
	
	Step3: Select The Statistical Test to Perform: ---> **Shapiro Test**
	
	Step4: Select the Significance Level **(SL = 0.05)**
	
	Step5: Find the **p-value of R&D Spend using Shapiro Test**
		SL = 0.05
		
		from scipy.stats import shapiro
		#from scipy.stats import anderson
		#from scipy.stats import normaltest
		
		corr, pvalue = shapiro(data['R&D Spend'])
		
		if pvalue >= SL:
		  print("Alternate Hypothesis passed (H1) -- R&D Spend is normally distributed")
		else:
		  print("Null Hypothesis passed (H0) -- R&D Spend is NOT normally distributed")
		
		#print("Confidence Level for R&D by Shapiro : {}".format(1-pvalue))
		
		Alternate Hypothesis passed (H1) -- R&D Spend is normally distributed
		Confidence Level for R&D by Shapiro : 0.8199481666088104

		import seaborn as sns %matplotlib inline sns.distplot(data['R&D Spend']) 

Non-parametric Test / Parametric Test Goal

Non-parametric Test ➔ goal of Non-parametric Test whether two features have any kind of relationship. If the two features **fail** normality test and then go for Non-Parametric Test **(feature and feature) (Use for feature elimination):**

1- Wilcoxon Test
2- Matt – Whitney U test
3- Kruskal-Wallis(H) Test
4- Friedman Test

Parametric Test ➔ goal of Non-parametric Test whether two features have any kind of relationship. There is a catch here, you should use **Parametric Test** if normality passes. The two features **pass** normality test and then go for Parametric Test **(feature and feature) (Use for feature elimination)**

	Non-parametric Test / Parametric Test Goal	SL = 0.05
		
		from scipy.stats import **wilcoxon**
		#from scipy.stats import mannwhitneyu
		#from scipy.stats import kruskal
		#from scipy.stats import friedmanchisquare
		
		corr, pvalue = wilcoxon(data['R&D Spend'], data['Administration'])
		
		if pvalue <= SL:
		  print("Alternate Hypothesis passed (H1) -- R&D Spend and Administration are Unequal")
		else:
		  print("Null Hypothesis passed (H0) -- R&D Spend and Administration are NOT Unequal")
		
		Alternate Hypothesis passed (H1) -- R&D Spend and Administration are Unequal(No multicollinearity)

	Parametric Test	SL = 0.05
		
		from scipy.stats import ttest_ind
		from scipy.stats import ttest_rel
		from scipy.stats import f_oneway
		
		#from scipy.stats import mannwhitneyu
		#from scipy.stats import kruskal
		#from scipy.stats import friedmanchisquare
		
		corr, pvalue = ttest_ind(data['R&D Spend'], data['Administration'])
		
		if pvalue <= SL:
		  print("Alternate Hypothesis passed (H1) -- R&D Spend and Administration are Unequal")
		else:
		  print("Null Hypothesis passed (H0) -- R&D Spend and Administration are NOT Unequal")
		
		Alternate Hypothesis passed (H1) -- R&D Spend and Administration are Unequal(No multicollinearity)

Chi-square Test

Use chi-square test in the following conditions: Feature and Label

a. Feature is Categorical and Label is Numerical
b. Feature is Categorical and Label is Categorical
c. Feature is Numerical and Label is Categorical

Goal is to test whether there exists any relationship between feature and label.
If Relationship exists maintain the feature else eliminate it.

	Checking the relationship between **State(Feature-Categorical) and Profit(Label - Numerical)**
	
	Step1: Prepare your data to make it compatible for Chi-square function
	
	Create Contingency Table
		c_t = pd.crosstab(data['State'], data['Profit'])
		
		# Apply CS Test
		
		from scipy.stats import chi2_contingency
		
		s,pvalue,a,b = chi2_contingency(c_t)
		
		if pvalue <= SL:
		  print("Alternate Hypothesis passed (H1) -- State and Profit have some form of relationship")
		else:
		  print("Null Hypothesis passed (H0) -- State and Profit DOESNOT have some form of relationship")
		
		Null Hypothesis passed (H0) -- State and Profit DOESNOT have some form of relationship