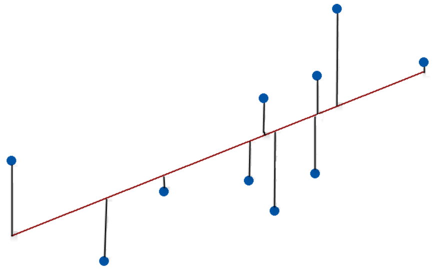
Harold's Statistics
ANOVA and Chi-Squared
Cheat Sheet

24 June 2022

Analysis of Variance (ANOVA) for Linear Regression

<table>
<tr><th>Term</th><th>Formula</th><th>Description</th></tr>
<tr><td><b>Problem Statement</b></td><td colspan="2"><i>How do we measure both the explained and unexplained variances?</i></td></tr>
<tr><td><b>Residual Sum of Squares (SSE)</b></td><td>$SSE = \sum e_i^2 = \sum (Y_i - \hat{Y}_i)^2$</td><td>Estimator errors</td></tr>
<tr><td><b>Residual Degrees of Freedom (df)</b></td><td>$df = n - p$</td><td>Number of regression parameters</td></tr>
<tr><td><b>Residual Mean Square (MSE)</b></td><td>$MSE = \frac{SSE}{n - p}$</td><td>Measures the amount of error in statistical models. 0 = no error.
</td></tr>
<tr><td><b>Residual Standard Error (s)</b></td><td>$s = \sqrt{MSE}$</td><td>Estimates the standard deviation of the residuals</td></tr>
<tr><td><b>Python</b></td><td colspan="2"><pre>
import pandas as pd
import statsmodels.api as sm
from statsmodels.formula.api import ols
scores = pd.read_csv('ExamScores.csv')

# Creates a linear regression model
results = ols('Exam4 ~ Exam1', data=scores).fit()
print(results.summary())

# The explained and unexplained variance can be obtained
from the analysis of variance table
aov_table = sm.stats.anova_lm(results, typ=2)
print(aov_table)
</pre><pre>
               sum_sq   df   F   PR(>F)
Exam1   217.166351   1.0 3.517655 0.066808
Residual 2963.333649 48.0   NaN   NaN
</pre></td></tr>
</table>

One-Way Analysis of Variance (One-Way ANOVA)

<table>
<tr><th>Test</th><th>Hypotheses</th><th>Research Question</th></tr>
<tr><td><b>Problem Statement</b></td><td colspan="2"><i>How do we determine whether a statistically significant difference exists among the means of three or more groups or populations?</i></td></tr>
<tr><td><b>One-Way ANOVA</b></td><td colspan="2">Tests for an association between a single categorical predictor variable and a response variable.</td></tr>
<tr><td><b>Factor</b></td><td colspan="2">A categorical predictor variable.</td></tr>
<tr><td><b>Level</b></td><td colspan="2">A possible value of a factor.</td></tr>
<tr><td><b>F-test</b></td><td>1. Hypotheses
$H_0 : \mu_1 = \mu_2 = \ldots = \mu_k$
$H_a : \mu_i \neq \mu_j$, for some $i \neq j$
2. F-test (F statistic)
$$F = \frac{between\ group\ variance}{within\ group\ variance}$$
3. p-value (Prob (F-statistic))
4. Significance level $\alpha = 0.05$</td><td>Assumes the k population means (μ) are independent.
If p-value < α, then reject H_0.
If p-value > α, then favor H_0.</td></tr>
<tr><td><b>Python 1</b></td><td colspan="2"><pre>
import pandas as pd
import scipy.stats as st
scores = pd.read_csv('ExamScores.csv')
# Statistics of each exam
exam1_score = scores[['Exam1']]
exam2_score = scores[['Exam2']]
exam3_score = scores[['Exam3']]
exam4_score = scores[['Exam4']]
print(st.f_oneway(exam1_score, exam2_score, exam3_score, exam4_score))
</pre><pre>
F_onewayResult(
statistic=array([3.85696089]),
pvalue=array([0.01034867]))
</pre></td></tr>
<tr><td><b>Python 2</b></td><td colspan="2"><pre>
import statsmodels.api as sm
import pandas as pd
from statsmodels.formula.api import ols
df = pd.read_csv('ExamScoresGrouped.csv')
mod = ols('Scores ~ Exam',data=df).fit()
aov_table = sm.stats.anova_lm(mod, typ=2)
print(aov_table)
</pre><pre>
             sum_sq   df   F   PR(>F)
Exam   2400.735   3.0 3.856961 0.010349
Residual 40666.220 196.0   NaN   NaN
</pre></td></tr>
</table>

Post-Hoc Tests

<table>
<tr><th>Test</th><th>Hypotheses</th><th>Research Question</th></tr>
<tr><td><b>Problem Statement</b></td><td colspan="2"><i>If the ANOVA null hypothesis is rejected, further analysis is required because the F-test does not determine which groups have different means.</i>
<i>Helps us identify groups that are significantly different than others.</i></td></tr>
<tr><td><b>Post-Hoc Analysis</b></td><td colspan="2">Determines which groups have different means, which group has the highest or lowest mean, and other relationships between the groups.</td></tr>
<tr><td><b>Tukey Honestly Significant Difference (HSD)</b></td><td colspan="2">A procedure that gives the 95% confidence intervals for the mean difference between pairwise groups and determines which mean difference is statistically significant.</td></tr>
<tr><td rowspan="2"><b>Python</b></td><td colspan="2"><pre>
import pandas as pd
from statsmodels.stats.multicomp import (pairwise_tukeyhsd,
MultiComparison)
df = pd.read_csv(ExamScoresGrouped.csv')
mod = MultiComparison(df['Scores'], df['Exam'])
print(mod.tukeyhsd())
</pre></td></tr>
<tr><td colspan="2"><pre>
Multiple Comparison of Means - Tukey HSD,FWER=0.05
==
group1 group2 meandiff lower   upper reject

Exam1 Exam2   -3.3   -10.7652 4.1652 False
Exam1 Exam3   -9.36   -16.8252 -1.8948 True
</pre></td></tr>
</table>

Chi-Square Tests for Comparing Categorical Variables

Goodness-of-Fit Test – Chi-Square		
Problem Statement	*The chi-square distribution is used to test how close the distribution of a population is to a theoretical distribution.* *The chi-squared test statistic measures how different the observed counts are compared to the expected counts, assuming the null hypothesis is true.*	
χ^2-test	1. Hypotheses (two-sided) H_0 : The random variable follows the expected distribution H_a : The random variable does not follow the expected distribution 2. χ^2-test statistic 3. p-value (Prob (χ^2-statistic)) 4. Significance level α = 0.05	For a chi-squared distribution If p-value < α, then reject H_0. If p-value > α, then favor H_0. If H_0 is rejected, insufficient evidence exists to conclude that the distribution does not follow the expected distribution.
Expected Frequencies for a Chi-Square	$E = np$	$p = proportion$ $n = sample\ size$
Chi-Square Test Statistic	$\chi^2 = \frac{(n-1)s^2}{\sigma^2}$ $\chi^2 = \sum \frac{(observed - expected)^2}{expected}$	Large χ^2 values are evidence against the null hypothesis, which states that the percentages of observed and expected match (as in, any differences are attributed to chance).
Degrees of Freedom	$df = k - 1$	$k = number\ of\ possible\ values\ (categories)\ for\ the\ variable\ under\ consideration$

Python	from scipy.stats import chisquare
	statistic, pvalue = chisquare([61, 17, 11, 15, 6], f_exp=[55, 25.3, 13.2, 11, 5.5])
	print(statistic)
	print(pvalue)	

	5.244137022397893 0.26315206062015767	

Independence Test – Chi-Square (2 Variables)

Problem Statement	*Determine whether two or more variables from a single population are independent by comparing the distributions of the variables over two or more categories.*	
χ^2**-test**	1. Hypotheses (two-sided) H_0 : The two variables are independent H_a : The two variables are not independent 2. χ^2-test statistic 3. p-value (Prob (χ^2-statistic)) 4. Significance level α = 0.05	A rule of thumb is that all expected individual cell counts should be at least five (5). Combine cells if less than 5. For a chi-squared distribution If p-value < α, then reject H_0, insufficient evidence exists to conclude that the two variables are not independent. If p-value > α, then favor H_0.
Expected Frequencies for a Chi-Square	$E = \frac{rc}{n}$	Contingency tables: $r = \#\ of\ rows$ $c = \#\ of\ columns$ $n = sample\ size$
Chi-Square Test Statistic	$\chi^2 = \frac{(O - E)^2}{E}$ $\chi^2 = \sum \frac{(observed\ count - expected\ count)^2}{expected\ count}$	
Degrees of Freedom	$df = (r - 1)(c - 1)$	$rc = number\ of\ possible\ values\ for\ the\ two\ variables\ under\ consideration$
Contingency Tables	A contingency table is constructed from the values of the variables and categories along the rows and columns. An expected cell count is calculated by multiplying the row total by the column total and dividing by the overall total.	

Observed			
	Category B		
Category A	**Group B1**	**Group B2**	**Total by Category A**
Group A1	5	8	**13**
Group A2	6	9	**15**
Group A3	7	10	**17**
Total by Category B	**18**	**27**	**45**

df = (3-1)(2-1) = 2

Expected			
	Category B		
Category A	**Group B1**	**Group B2**	**Total by Category A**
Group A1	$\frac{13 \bullet 18}{45} = 5.2$	$\frac{13 \bullet 27}{45} = 7.8$	**13**
Group A2	$\frac{15 \bullet 18}{45} = 6$	$\frac{15 \bullet 27}{45} = 9$	**15**
Group A3	$\frac{17 \bullet 18}{45} = 6.8$	$\frac{17 \bullet 27}{45} = 10.2$	**17**
Total by Category B	**18**	**27**	**45**

Python

```
import numpy as np
from scipy.stats import chi2_contingency

# Construct a contingency table
parole = np.array([[405,1422], [240,470], [151,275]])

# Calculate the test statistic, p-value, df, and expected counts
chi2, p, df, ex = chi2_contingency(parole)
print(chi2, p, df, ex)
```

```
X2-test: 53.87860692066112
p-value: 1.9971429926442894e-12
df: 2
[[ 490.81741478 1336.18258522]
 [ 190.73911576  519.26088424]
 [ 114.44346946  311.55653054]]
```

Homogeneity Test – Chi-Square (Multiple Variables)

Problem Statement

Determines if two or more populations (or subgroups of a population) have the same unknown distribution of a single categorical variable.

χ^2-test

1. Hypotheses (two-sided)

   H_0: The distribution of the frequency of one variable is the same across all sampled populations

   $p_{1,1} = p_{1,2} = \ldots = p_{1,J}$
   $p_{2,1} = p_{2,2} = \ldots = p_{2,J}$
   $\ldots$
   $p_{I,.1} = p_{I,2} = \ldots = p_{I,J}$

   $p_{1,1} = p_{2,1} = \ldots = p_{I,1}$
   $p_{1,2} = p_{2,2} = \ldots = p_{I,2}$
   $\ldots$
   $p_{,1,J} = p_{2,J} = \ldots = p_{I,J}$

   H_a: At least one of the probability statements is false.
2. χ^2-test statistic
3. p-value (Prob (χ^2-statistic))
4. Significance level α = 0.05

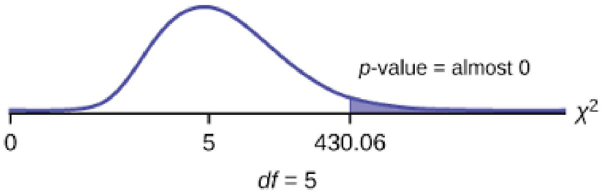
i = 1, 2,…, I represent the categories of the first variable.
j = 1,2,…, J represent the categories of the second variable.

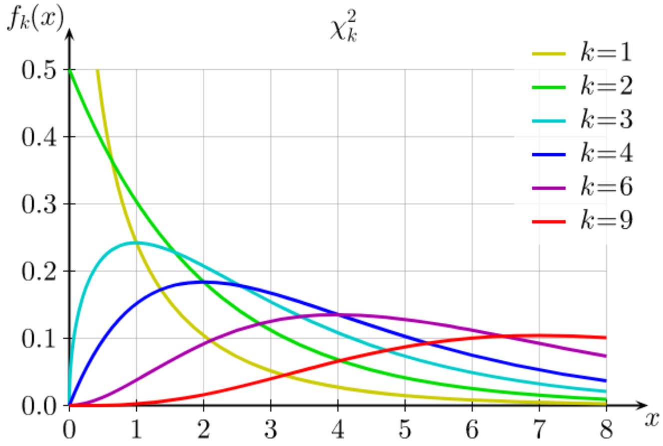
For a chi-squared distribution
If p-value < α, then reject H_0.
If p-value > α, then favor H_0.

If H_0 is rejected, conclude that the distribution of one variable is not the same across categories of the other variable.

Chi-Square Test Statistic

The test for homogeneity is performed in the same way as the test for independence.





Python

```
import numpy as np
from scipy.stats import chi2_contingency

z = np.array([[551, 580], [244, 289], [387, 503], [452, 618],
[443,742]])

chi2, p = chi2_contingency(z)
print(chi2)
print(p)
```

```
X2-test: 32.2443
p-value: 1.70526e-6
```