Hypotheses Testing

In [1]:
```python
# Importing pandas for dataframe operations
import pandas as pd

# Importing scipy.stats for statistical computations
import scipy.stats
import statsmodels.stats
```

Reading CSV file

In [2]:
```python
cars_data=pd.read_csv('cars_sampled.csv' )
```

Creating copy

In [3]:
```python
cars=cars_data.copy()
```

Working range of data

In [4]:
```
cars = cars[
        (cars.yearOfRegistration <= 2018)
      & (cars.yearOfRegistration >= 1950)
      & (cars.price >= 100)
      & (cars.price <= 150000)
      & (cars.powerPS >= 10)
      & (cars.powerPS <= 500)]
```

All hypotheses will be checked on a sample of data

One sample test for mean

Three years back, the average price of a used car was 6000 \$.
Has it changed now?

Hypotheses Testing Steps

Hypotheses	H0 : μ = 6000 HA : μ ≠ 6000
Sample Statistics	$\bar{x}$ "s" being used as an estimator of " σ "
Test Statistics	"t" value-?
Critical Values	?
Max Uncertainty α	0.05
Computed Uncertainty p	?
Decision on H0	?

Arriving at a sub sample from 'cars' data

In [5]:

```
sample_size=1000
sample1=cars.sample(sample_size,random_state=0)
```

Postulated mean and sample mean

In [6]:
```
pos_mean = 6000
```

In [7]:
```
print(sample1['price'].mean())
```

```
6188.337
```

Importing the package for one sample t-test

In [8]:
```
from scipy.stats import ttest_1samp
```

In [9]:
```
statistic,pvalue = ttest_1samp(sample1['price'],pos_mean)
print(statistic,pvalue)
```

```
0.8148683326967585 0.41534189398889065
```

Calculating the degrees of freedom

In [10]:

```
# No. of observations/records in data
n = len(sample1['price'])
# Degrees of freedom= n-1
df = n-1

print(n,df)
```

```
1000 999
```

Significance level

In [11]:

```
alpha=0.05
```

Importing the package for t distribution

In [12]:
```
from scipy.stats import t
```

In [13]:
```
# Critical values from standard distribution
cv = t.ppf([alpha/2,1-alpha/2],df)
print(cv)
```

```
[-1.96234146  1.96234146]
```

Hypotheses Testing Steps

Hypotheses	H0 : $\mu = 6000$ HA : $\mu \neq 6000$
Sample Statistics	$\bar{x}$ = 6188.34 "s" being used as an estimator of " σ "
Test Statistics	"t" value=0.815
Critical Values	[-1.96,+1.96]
Max Uncertainty α	0.05
Computed Uncertainty p	0.415
Decision on H0	Do Not Reject Null Hypothesis Conclude $\mu = 6000$

One sample test for proportion

Three years back, % of used car with automatic transmission were 23%. Has it changed now?

Hypotheses Testing Steps

Hypotheses	H0 : $\pi = 0.23$ HA : $\pi \neq 0.23$
Sample Statistics	$\hat{p}$ "H0" is used to compute "σ"
Test Statistics	"z" value-?
Critical Values	?
Max Uncertainty α	0.05
Computed Uncertainty p	?
Decision on H0	?

We will be using 'sample1' again

Importing the package for one sample z-test

In [14]:
```
from statsmodels.stats.proportion import proportions_ztest
```

In [15]:

```
# No.of gearbox='automatic'
count = sample1['gearbox'].value_counts()[1]

# Total no. of observations
nobs = len(sample1['gearbox'])

# Hypothesized value
p0 = 0.23
```

In [16]:

```
# In the sample
sample1['gearbox'].value_counts()/nobs
```

Out[16]:

```
manual       0.771
automatic    0.214
Name: gearbox, dtype: float64
```

$$\sigma_p = \sqrt{\frac{\pi(1-\pi)}{n}}$$

$$z = \frac{\hat{p} - \pi}{\sigma_p}$$

In [17]:
```
statistic_oneprop,pvalue_oneprop = proportions_ztest(count=count, nobs=nobs, value=p0,
                                                     alternative='two-sided', prop_var=False)
print(statistic_oneprop,pvalue_oneprop)
```

```
-1.233678008148831 0.21732291189942932
```

In [18]:
```
# Importing normal distribution
from scipy.stats import norm

# Critical values
cv_norm = norm.ppf([alpha/2,1-alpha/2])
print(cv_norm)
```

```
[-1.95996398  1.95996398]
```

Hypotheses Testing Steps

Hypotheses	H0 : π = 0.23 HA : π ≠ 0.23
Sample Statistics	$\hat{p}$ =0.214
Test Statistics	“z” value= -1.23
Critical Values	[-1.96, +1.96]
Max Uncertainty α	0.05
Computed Uncertainty p	0.22
Decision on H0	Do Not Reject Null Hypothesis Conclude π = 0.23

Two sample test for means

Is the mean price of cars that have run 30000 - 60000 KM, the same as that for cars that have run 70000 - 90000 KM?

Hypotheses Testing Steps

Hypotheses	$\mathrm{H}_0 : \mu_1 = \mu_2$ $\mathrm{H}_\mathrm{A} : \mu_1 \neq \mu_2$
Sample Statistics	$\bar{x}$ "s" being used as an estimator of " σ "
Test Statistics	Depends on whether the two groups have equal or unequal variance
Critical Values	?
Max Uncertainty α	0.05
Computed Uncertainty p	?
Decision on H0	?

We first need to test whether the variance in price of cars that have run 30000 - 60000 KM, the same as the variance in price of cars that have run 70000 - 90000 KM?

Subsetting records based on kilometer limits and drawing 500 samples from each

In [19]:
```
km_70000_90000=cars[(cars.kilometer <= 90000) & (cars.kilometer >= 70000)]
km_30000_60000=cars[(cars.kilometer <= 60000) & (cars.kilometer >= 30000)]
```

In [20]:
```
sample_70000_90000=km_70000_90000.sample(500,random_state=0)
sample_30000_60000=km_30000_60000.sample(500,random_state=0)
```

Sample variance

In [21]:
```
print(sample_70000_90000.price.var())
print(sample_30000_60000.price.var())
```

```
86753098.35060121
155442577.94620845
```

Sample mean

In [22]:
```
print(sample_70000_90000.price.mean())
print(sample_30000_60000.price.mean())
```

```
9450.59
14515.678
```

Computing the F statistic

In [23]:
```
from scipy.stats import f
F=sample_70000_90000.price.var()/sample_30000_60000.price.var()
print(F)
```

```
0.5581038316324275
```

Calculating the degrees of freedom for the two samples

In [24]:
```
df2=len(sample_70000_90000)-1
df1=len(sample_30000_60000)-1
```

In [25]:
```
scipy.stats.f.cdf(F, df1, df2)
```

Out[25]:
```
5.0498268005416406e-11
```

In [26]:
```
f.ppf([alpha/2,1-alpha/2],df1, df2)
```

Out[26]:
```
array([0.83888578, 1.1920574 ])
```

Hypotheses Testing Steps

Hypotheses	$\mathrm{H}_0 : \sigma_1^2 = \sigma_2^2$ $\mathrm{H}_\mathrm{A} : \sigma_1^2 \neq \sigma_2^2$
Sample Statistics	Variance 30000-60000=155442577.95 70000-90000=86753098.35
Test Statistics	F statistic=0.56
Critical Values	[0.84,1.19]
Max Uncertainty α	0.05
Computed Uncertainty p	$5.05 * 10^{-11}$
Decision on H0	Reject H0 $\sigma_1^2 \neq \sigma_2^2$ Unequal variances

Welch t test for unequal variances

Hypotheses Testing Steps

Hypotheses	$\mathrm{H}_0 : \mu_1 = \mu_2$ $\mathrm{H}_\mathrm{A} : \mu_1 \neq \mu_2$
Sample Statistics	$\bar{x}$ “s” being used as an estimator of “ σ ”
Test Statistics	Welch t test for unequal variance
Critical Values	?
Max Uncertainty α	0.05
Computed Uncertainty p	?
Decision on H0	?

In [27]:
```
from scipy.stats import ttest_ind
statistic_twomean,pvalue_twomean=ttest_ind(sample_30000_60000.price,sample_70000_90000.p
rice,equal_var=False)
print(statistic_twomean,pvalue_twomean)
```

```
7.277610434526923 7.258473522297715e-13
```

To get critical values we need degrees of freedom

$$df \approx \frac{\left(\frac{s_1^2}{N_1}+\frac{s_2^2}{N_2}\right)^2}{\frac{s_1^4}{N_1^2 df_1}+\frac{s_2^4}{N_2^2 df_2}}$$

In [28]:
```
N1=len(sample_30000_60000)
N2=len(sample_70000_90000)
s12=sample_30000_60000.price.var()
s22=sample_70000_90000.price.var()
df=(((s12/N1)+(s22/N2))**2)/((((s12/N1)**2)/(N1-1))+(((s22/N2)**2)/(N2-1)))
print(df)
```

```
923.7016134521467
```

In [29]:
```
cv_t = t.ppf([alpha/2,1-alpha/2],df)
print(cv_t)
```

```
[-1.96253552  1.96253552]
```

Hypotheses Testing Steps

Hypotheses	$\mathrm{H}_0 : \mu_1 = \mu_2$ $\mathrm{H}_\mathrm{A} : \mu_1 \neq \mu_2$
Sample Statistics	$\bar{x}$ 30000-60000=14515.68 dollar 70000-90000=9450.59 dollar
Test Statistics	Welch t test for unequal variance t statistic = 7.28
Critical Values	[-1.96,+1.96]
Max Uncertainty α	0.05
Computed Uncertainty p	$7.26^{*}10^{-13}$
Decision on H0	Reject H0 μ1 ≠ μ2

Two sample test for proportion

Are the proportion petrol cars in two different time periods 2009 – 2013, and 2014 – 2018, different?

Hypotheses Testing Steps

Hypotheses	$\mathrm{H}_0 : \pi_1 = \pi_2$ $\mathrm{H}_\mathrm{A} : \pi_1 \neq \pi_2$
Sample Statistics	$\hat{p}$ Pooled estimate is used to compute “σ”
Test Statistics	“z” value-?
Critical Values	?
Max Uncertainty α	0.05
Computed Uncertainty p	?
Decision on H0	?

$$\hat{p} = \frac{n_1 P_1 + n_2 P_2}{n_1 + n_2}$$

$$S_{P_1 - P_2} = \sqrt{\hat{p}(1 - \hat{p}) \left[\frac{1}{n_1} + \frac{1}{n_2} \right]}$$

$$Z = \frac{P_1 - P_2}{S_{P1 - p2}}$$

Subsetting records based on year and drawing 1000 samples from each

In [30]:
```
year_2014_2018=cars[(cars.yearOfRegistration <= 2018) & (cars.yearOfRegistration >= 2014
)]
year_2009_2013=cars[(cars.yearOfRegistration <= 2013) & (cars.yearOfRegistration >= 2009
)]
```

In [31]:
```
sample_2014_2018=year_2014_2018.sample(1000,random_state=0)
sample_2009_2013=year_2009_2013.sample(1000,random_state=0)
```

In [32]:
```
from statsmodels.stats.proportion import proportions_ztest
count = [(sample_2014_2018['fuelType']=='petrol').sum(),(sample_2009_2013['fuelType']==
'petrol').sum()]
nobs = [len(sample_2014_2018),len(sample_2009_2013)]
```

In [33]:
```
print(count[0]/nobs[0])
print(count[1]/nobs[1])
```

```
0.494
0.506
```

```
In [34]: statistic,pvalue = proportions_ztest(count=count, nobs=nobs, value=0, \
                                              alternative='two-sided', prop_var=False)
```

```
In [35]: print(statistic,pvalue)
```

```
-0.53665631459995 0.5915050369949162
```

```
In [36]: cv = norm.ppf([alpha/2,1-alpha/2])
         print(cv)
```

```
[-1.95996398  1.95996398]
```

Hypotheses Testing Steps

Hypotheses	$\mathrm{H}_0 : \pi_1 = \pi_2$ $\mathrm{H}_\mathrm{A} : \pi_1 \neq \pi_2$
Sample Statistics	$\hat{p}$ 2009-2013=0.506 2014-2018=0.494
Test Statistics	"z" value= -0.54
Critical Values	[-1.96,+1.96]
Max Uncertainty α	0.05
Computed Uncertainty p	0.59
Decision on H0	Do Not Reject Null Hypothesis Conclude π1 = π2

Chi-square test of independence

Is vehicleType dependent on fuelType?

In [37]:

```python
# Cross table between fuelType and vehicleType
cross_table=pd.crosstab(cars['fuelType'],cars['vehicleType'])
```

In [38]:

```python
cross_table
```

Out[38]:

vehicleType	bus	cabrio	coupe	limousine	others	small car	station wagon	suv
fuelType								
cng	31	1	1	6	2	11	16	0
diesel	2257	195	324	3446	159	839	4266	1120
electro	0	0	0	1	0	9	0	0
hybrid	0	0	2	19	1	6	5	3
lpg	74	35	47	218	3	64	137	92
other	0	1	1	3	0	0	1	0
petrol	1183	2500	1831	7755	142	8020	3406	592

In [39]:
```
scipy.stats.chi2_contingency(cross_table)
```

Out[39]:
```
(7987.74154009857,
 0.0,
 42,
 array([[6.20888603e+00, 4.78495815e+00, 3.86369607e+00, 2.00505860e+01,
         5.37694784e-01, 1.56737154e+01, 1.37155956e+01, 3.16486800e+00],
        [1.15101790e+03, 8.87046800e+02, 7.16261069e+02, 3.71702480e+03,
         9.96791243e+01, 2.90563024e+03, 2.54262939e+03, 5.86710676e+02],
        [9.13071475e-01, 7.03670316e-01, 5.68190599e-01, 2.94861558e+00,
         7.90727624e-02, 2.30495815e+00, 2.01699936e+00, 4.65421764e-01],
        [3.28705731e+00, 2.53321314e+00, 2.04548616e+00, 1.06150161e+01,
         2.84661945e-01, 8.29784932e+00, 7.26119768e+00, 1.67551835e+00],
        [6.11757888e+01, 4.71459111e+01, 3.80687701e+01, 1.97557244e+02,
         5.29787508e+00, 1.54432196e+02, 1.35138957e+02, 3.11832582e+01],
        [5.47842885e-01, 4.22202189e-01, 3.40914359e-01, 1.76916935e+00,
         4.74436574e-02, 1.38297489e+00, 1.21019961e+00, 2.79253059e-01],
        [2.32184945e+03, 1.78936325e+03, 1.44485187e+03, 7.49803457e+03,
         2.01074127e+02, 5.86127807e+03, 5.12902766e+03, 1.18352100e+03]]))
```

In [40]:

```python
# Explain output
pd.crosstab(cars['fuelType'],cars['vehicleType'],margins=True)
```

Out[40]:

vehicleType	bus	cabrio	coupe	limousine	others	small car	station wagon	suv	All
fuelType									
cng	31	1	1	6	2	11	16	0	68
diesel	2257	195	324	3446	159	839	4266	1120	12606
electro	0	0	0	1	0	9	0	0	10
hybrid	0	0	2	19	1	6	5	3	36
lpg	74	35	47	218	3	64	137	92	670
other	0	1	1	3	0	0	1	0	6
petrol	1183	2500	1831	7755	142	8020	3406	592	25429
All	3545	2732	2206	11448	307	8949	7831	1807	38825

$$E_{ij} = \frac{R_i C_j}{N}$$

$$\chi^2 = \sum \frac{(O - E)^2}{E}$$

In [41]:

```python
68*3545/38825
```

Out[41]:

```
6.20888602704443
```

In [42]:
```
df=(cross_table.shape[0]-1)*(cross_table.shape[1]-1)
print(df)
```

```
42
```

In [43]:
```
from scipy.stats import chi2
chi2.ppf(q=[alpha/2,1-alpha/2],df=42)
```

Out[43]:
```
array([25.99866197, 61.77675581])
```

Hypotheses Testing Steps

Hypotheses	H0: vehicleType dependent on fuelType HA:vehicleType is not dependent on fuelType
Test Statistics	χ^2 = 7987.7
Critical Values	[25.99,61.78]
Max Uncertainty α	0.05
Computed Uncertainty p	0
Decision on H0	Reject Null Hypothesis Conclude vehicleType is not dependent on fuelType