Significance of individual predictor variables: Wald test vs Likelihood ratio test

1. What do the Wald test and the Likelihood ratio test measure?
    - Both assess the importance of individual independent variables to predicting the outcoming variable.
    - The null hypothesis for both is that the smaller model is the "true" model, a large test statistics indicate that the null hypothesis is false.
    - Both tests use the likelihood of the models being compared to assess their fit. The likelihood is the probability the data given the parameter estimates. The goal of a model is to find values for the parameters (coefficients) that maximize value of the likelihood function, that is, to find the set of parameter estimates that make the data most likely. Many procedures use the log of the likelihood, rather than the likelihood itself, because it is easier to work with. The log likelihood (i.e., the log of the likelihood) will always be negative, with higher values (closer to zero) indicating a better fitting model.

2. How do the tests compare?
    - The advantage of the Wald test is that it approximates the LR test, but require that only one model be estimated. The Wald is asymptotically equivalent to the LR test, that is, as the sample size becomes infinitely large, the value of the Wald test statistics will become increasingly close to the test statistic from the LR test. In finite samples, the two tests will tend to generate somewhat different test statistics, but will generally come to the same conclusion.
    - The Wald test is the most widely used test. However, the Wald statistic can be unreliable for small sample sizes and/or large coefficients, so it may be better to stick to the likelihood-ratio test. The only justification given for using the Wald statistic is that it is computationally easy and is given automatically in the output of most statistical computer packages.
    - When computing power was much more limited, and many models took a long time to run, being able to approximate the LR test using a single model was a fairly major advantage. Today, for most of the models researchers are likely to want to compare, computational time is not an issue, and it is generally recommend to use the likelihood ratio test in most situations.
3. Wald test: This test assesses the significance of individual variable, by testing whether each coefficient is significantly different from zero.
    - The Wald test is therefore similar to the hypothesis tests typically printed in regression output (standard errors). The difference is that the Wald test can be used to test multiple parameters simultaneously, while the tests typically printed in regression output only test one parameter at a time.
    - The following null and alternative hypotheses are used for this test:

      H0: Some set of predictor variables are all equal to zero.

      HA: Not all predictor variables in the set are equal to zero.

      If we fail to reject the null hypothesis (p value > .05), this means that the regression coefficients for the predictor variables are both equal to 0. Consequently, we can drop the specified set of predictor variables from the model because they don't offer a statistically significant improvement in the fit of the model.

- The procedure corresponds to backward elimination in multiple regression, that is, the least significant parameter is removed first, residuals are computed, then next least significant parameter is removed, and so on until a set is obtained that is simultaneously not significant.

```
import statsmodels.formula.api as smf
import pandas as pd
import io

#LOAD DATA SET

#Define URL where dataset is located
url = "XXX"

#Read data set
data = pd.read_csv(url)

#View regression model summary
results.summary()

#Perform Wald Test to determine if the regression coefficients for predictor
variables (calling them 'a' and 'b' as an example) are both zero
print(results.wald_test('(a = 0, b = 0)'))

F test: F=array([[0.91125429]]), p=0.41403001184235005, df_denom=27, df_num=2

#The output shows a p-value of 0.414. Since this p-value is more than .05, we
fail to reject the null hypothesis of the Wald test. This means we can assume
the regression coefficients for the predictor variables “a” and “a” are both
equal to zero. We can drop these terms from the model since they don’t
statistically significantly improve the overall fit of the model.
```

Algebraically speaking -

$$\text{Wald statistic} = \frac{b_1^2}{SE_1^2}$$

where

- b_1 is the coefficient for explanatory variable 1,
- SE_1 is its standard error

4. Likelihood ratio test: If you're comparing nested models like we are (e.g., adding or removing variables from a model), you can use the likelihood ratio test to assess whether the added variables significantly improve model fit. A nested model is simply one that contains a subset of the predictor variables in the overall regression model.
    - The LR test is performed by estimating two models and comparing the fit of one model to the fit of the other. Removing predictor variables from a model will almost always make the model fit less well (i.e., a model will have a lower log likelihood), but it is necessary to test whether the observed difference in model fit is statistically significant. The LR test does this by comparing the

log likelihoods of the two models, if this difference is statistically significant, then the less restrictive model (the one with more variables) is said to fit the data significantly better than the more restrictive model.

- THE LR test uses the following null and alternative hypotheses:

  H0: The full model and the nested model fit the data equally well. Thus, you should use the nested model.

  HA: The full model fits the data significantly better than the nested model. Thus, you should use the full model.

  If the p-value of the test is below a certain significance level (e.g. 0.05), then we can reject the null hypothesis and conclude that the full model offers a significantly better fit.

```
from sklearn.linear_model import LinearRegression
import statsmodels.api as sm
import pandas as pd
import scipy

#LOAD DATA SET

#Define URL where dataset is located
url = "https://example.com"

#Read data set
data = pd.read_csv(url)


#FULL MODEL: FIT & CALCULATE LOG-LIKELIHOOD

#Define outcome variable
y1 = data['y']

#Define predictor variables
x1 = data[['xa', 'xb', 'xc', 'xd']]

#Add constant to predictor variables
x1 = sm.add_constant(x1)

#Fit regression model
full_model = sm.OLS(y1, x1).fit()

#Calculate log-likelihood of model
full_ll = full_model.llf

print(full_ll)

#NESTED/REDUCED MODEL: FIT & CALCULATE LOG-LIKELIHOOD

#Define outcome variable
```

```
y2 = data['y']

#Define predictor variables
x2 = data[['xa', 'xb']]

#Add constant to predictor variables
x2 = sm.add_constant(x2)

#Fit regression model
reduced_model = sm.OLS(y2, x2).fit()

#Calculate log-likelihood of model
reduced_ll = reduced_model.llf

print(reduced_ll)

#PERFORM LOG-LIKELIHOOD TEST

#Calculate likelihood ratio Chi-Squared test statistic
LR_statistic = -2*(reduced_ll-full_ll)

print(LR_statistic)

#Calculate p-value of test statistic using p degrees of freedom (for number of
parameters, f.e. 2)
p_val = scipy.stats.chi2.sf(LR_statistic, p)

print(p_val)

#For instance, the Chi-Squared test-statistic could be 2.0902 and the
corresponding p-value is 0.3517. Since this p-value is higher than .05, we
would fail to reject the null hypothesis. This means the full model and the
nested model fit the data equally well. Thus, we should use the nested model
because the additional predictor variables in the full model don't offer a
significant improvement in fit.
```

Algebraically speaking -

$$G = -2\ln\left[\frac{L_{reduced}}{-L_{full}}\right]$$

where

- $L_{reduced}$ is the log likelihood for the model without the predictor,
- L_{full} is the log likelihood for the full model