Empirical exercise – The HCE, tests of heteroskedasticity, FGLS estimator

1. Aim of the exercise

In this exercise we analyse the factors that affect consumer memory for television advertising. Using data on consumer memory, we first estimate a standard linear model. We suspect that the model is subject to heteroskedasticity. We compare the usual standard error estimator of the OLS coefficient estimates with the heteroscedasticity-consistent estimator (HCE) and check how they differ. We then use formal tests to show that the standard regression model of interest is in fact subject to heteroskedasticity and that the analysis could be carried out using the generalised linear model that allows for heteroskedasticity. We then use the feasible generalised least squares (FGLS) estimator to estimate this model.

The empirical context is as follows. '*Rik G. M. Peters and Tammo H. A. Bijmolt, 1997. Consumer memory for television advertising: a field study of duration, serial position, and competition effects. Journal of Consumer Research, 23 (4), 362-372*' analyse how television advertising affect consumer memory for the advertised product. To this purpose data were collected from February 1975 to February 1992. In particular, Dutch consumers are asked to watch blocks of commercials at given dates, at a given channel, in the evenings at around 7 or 8 pm, for a duration of 10 to 30 minutes. After watching a block of commercials, a personal interview is conducted with the consumer. In particular, the consumer was asked which television commercials he can remember having seen in the respective block of commercials. In the answer, the brand name and the product category should be mentioned correctly. In some cases no aid is provided to the consumer to help to recall the commercials, in others aid is provided. Such interviews are held for 224 blocks of commercials which included 2,677 commercials shown on almost all public channels. The average sample size per block was about 175, and hence the data is based on about 39,200 respondents. In this exercise observations for 2,677 commercials are used.

Descriptions of the variables contained in the data are as follows. 'unaid' is the fraction of respondents who mention the commercial in the unaided recall question among all respondents who watched the block the commercial was part of. 'dur' is the duration of the commercial in seconds. 'ncb' is the number of commercials in the block. 'rank' is the rank of the commercial in the block (1 if first commercial, . . . , ncb if last). 'year' is the year in which the commercial was broadcasted.

Our aim is to explain unaid with dur, ncb, rank, year, and an intercept term, using the standard or the generalised linear model.

2. Load the data

Load the data in .mat format to the memory.

```
clear;
load 'M:\exercisehcefgls.mat';
clearvars aid;
```

3. Exploratory graphical analysis

Using the code presented at the end of the section, produce a scatter plot of the dependent variable against an independent variable. Overlay this plot with the least squares line. Inspect the spread of the points at given values of the independent variable. Is the spread constant across the values of the independent variable? If yes, what do you conclude? Repeat this analysis with each regressor, that is with `dur`, `ncb`, `rank`, and `year`.

```
scatter(dur,unaid,'filled','black') % Repeat the plot with 'ncb', 'rank', 'year'.
hold on
set(lsline,'color','blue','LineWidth',2)
hold off
title('Fig 1. Scatter plot of the dep. var. against an ind. var. overlaid with
the least squres line')
legend('Scatter Plot','Fitted Line');
```

4. Create the systematic component of the regression equation and estimate the model

Using the code presented at the end of the section create the systematic component of the regression equation. Assume a standard linear regression model, and produce OLS estimation results and related statistics using the supplied function `exercisefunctionlssrobust`. This function will produce output and store it in the structure array `LSS`.

Obtain the residuals stored in the structure array, and name them as `u_hat` for future reference.

```
y = unaid;
N = length(y);
x_0 = ones(N,1);
X = [x_0 dur ncb rank year];
LSS = exercisefunctionlssrobust(y,X);
u_hat = LSS.u_hat;
```

5. Obtain the OLS coefficient estimates

Obtain the OLS coefficient estimates stored in the structure array `LSS`, and name them as `B_hat`. Assume that strict exogeneity holds. We could interpret the coefficient estimate of `dur` as follows. The fraction of the respondents who recall a certain commercial increases by 0.567 percentage points if the duration of a given commercial increases by 1 second, on average, holding other factors constant. Considering the effects of the other independent variables, we find that the fraction of the respondents decreases with increasing number of commercials in the block, with the rank of the commercial in the block, and with the year when the commercial is broadcasted.

```
B_hat = LSS.B_hat;
```

6. Obtain the standard error estimates of the OLS coefficient estimates

Obtain the standard error estimates of the OLS coefficient estimates assuming that the errors of the regression are homoskedastic. In MATLAB these estimates can be calculated as follows. Start with the expression `LSS.u_hat'*LSS.u_hat`. This is the residual sum of squares. Next, consider the expression `1/(LSS.N-LSS.K)*LSS.u_hat'*LSS.u_hat`. This is the estimator of the variance of the regression error. Finally, consider the expression `1/(LSS.N-LSS.K)*(LSS.u_hat'*LSS.u_hat)*inv(X'*X)`. This is the usual variance-covariance estimator. The square root of it is the standard error estimator.

The supplied function calculates the standard error estimates and stores them in the vector array `LSS.B_hat_SEE`. Name this vector array as `B_hat_SEE`.

Now take a moment to express the variance-covariance estimator in an alternative form. The purpose of doing this will get clear in the next section. The variance-covariance estimator is `1/(LSS.N-LSS.K)*(LSS.u_hat'*LSS.u_hat)*inv(X'*X)`. Post multiply with `X'*X*inv(X'*X)` to obtain `1/(LSS.N-LSS.K)*(LSS.u_hat'*LSS.u_hat)*inv(X'*X)*X'*X*inv(X'*X)`. As the term `1/(LSS.N-LSS.K)*(LSS.u_hat'*LSS.u_hat)` is a scalar, we can move it across the matrices to obtain `inv(X'*X)*X'*1/(LSS.N-LSS.K)*(LSS.u_hat'*LSS.u_hat)*X*inv(X'*X)`. Finally, multiply the scalar `1/(LSS.N-LSS.K)*(LSS.u_hat'*LSS.u_hat)` with the identity matrix `eye(N)` which gives a diagonal matrix that contains this scalar along the diagonal. This leads to `inv(X'*X)*X'*(1/(LSS.N-LSS.K)*LSS.u_hat'*LSS.u_hat.*eye(N))*X*inv(X'*X)`. In MATLAB if you consider the two forms of the variance-covariance estimator, marked in blue and red colours, you can verify that they return the same values.

```
B_hat_SEE = LSS.B_hat_SEE;
```

7. Obtain the standard error estimates of the OLS coefficient estimates robust to heteroskedasticity

Recall how the standard variance-covariance estimator is derived. In a step in this derivation it is assumed that the errors are homoskedastic. If we discard this assumption, the standard variance-covariance estimator is not justified. If we discard this assumption and allow the errors to be heteroskedastic, the valid estimator can be calculated in MATLAB as `inv(X'*X)*X'*(LSS.u_hat.*LSS.u_hat.*eye(N))*X*inv(X'*X)`. This estimator is called the heteroskedasticity-consistent variance-covariance estimator (HCE). In the expression, note the dot operator `.*`. It consists of a dot, and the standard multiplication operator. It performs element-wise multiplication on matrices. The vector operation `LSS.u_hat.*LSS.u_hat` produces a vector where each residual in the residual vector `LSS.u_hat` is multiplied by itself.

Compare the robust variance-covariance estimator just stated with the standard variance-covariance estimator given in the preceding section. Both are marked in red colour above. The two estimators differ in the terms `1/(LSS.N-LSS.K)*LSS.u_hat'*LSS.u_hat.*eye(N)` and `LSS.u_hat.*LSS.u_hat.*eye(N)`. Inspect the output two terms return in MATLAB. The former produces a diagonal matrix where across the diagonal the variance of the residuals are the same. Hence, the standard variance-covariance estimator treats the variance of the residual constant across all observations. The latter produces a diagonal matrix where across the diagonal the variance of the residuals differ. Hence, the robust variance-covariance estimator accounts for the variance of the residual of each observation. This means it accounts for heteroskedasticity. It does not require the explicit functional form of heteroskedasticity which we usually do not know! The comparison of the two estimators also shows that if there is no heteroskedasticity the two estimators are the same because in this case the diagonal matrices considered in these estimators are the same and this is because the estimators of the error variance in the diagonals of these matrices are the same. This means that in empirical work you cannot go wrong if you always use the robust estimator.

Note that the heteroskedasticity-consistent variance-covariance estimator is a consistent estimator, but not unbiased. Therefore keep in mind that the robust standard errors are only appropriate when the sample size is large. Also note that the heteroskedasticity-consistent estimator is sometimes multiplied with the factor `LSS.N/(LSS.N-LSS.K)`. This is a degrees of freedom adjustment. Stata considers it and therefore we also consider it in the function file. See a discussion on this in the Stata manual of the regress function, or on page 200 in Davidson and MacKinnon (1999).

The supplied function calculates the heteroskedasticity-consistent standard error estimates and stores them in the vector array `LSS.B_hat_SEE_robust`. Name this vector array as `B_hat_SEE_robust`. Compare `B_hat_SEE` from the preceding section with `B_hat_SEE_robust`. What do you conclude? Can you use the t statistic that uses the usual standard error estimator? The robust standard errors are larger compared to the usual standard errors. This signals that heteroskedasticity is in play and that we should use the robust standard error estimator. The t statistic depends on the standard error estimator. Since heteroskedasticity seems to be of concern, the t statistic is valid only if the robust standard error estimator is used. It seems that heteroskedasticity is present because there is unexplained heterogeneity left in the error of the regression. In the next section we carry out formal tests to prove or disprove this.

```
B_hat_SEE_robust = LSS.B_hat_SEE_robust;
```

8. Test for heteroskedasticity of unknown form

If there is heteroskedasticity, $\mathrm{Var}\left[u_i \mid X_i\right] = \mathrm{E}\left[u_i^2 \mid X_i\right]$ should depend on X_i or on any function of X_i. This conditional expectation calls for a regression of u_i^2 on X_i or any function of X_i. u_i is not observed but it can be estimated with the OLS residual $\hat{u}_i$. In the regression of $\hat{u}_i^2$ on a general function of X_i, joint significance of the coefficients provides statistical evidence that the errors of the assumed model are heteroskedastic.

Carry out the test using the code presented at the end of the section. Create the dependent variable `u_hat_sq`. Let the systematic component of the regression be a general function of the regressors themselves, the cross products of them, and the square of each regressor. Let the vector arrays `R`, `C`, and `S` to contain these three sets of regressors. Use the supplied function to carry out the OLS regression. Use the R squared from this regression to construct the Lagrange multiplier test statistic. This statistic follows a Chi-squared distribution with `K-1` degrees of freedom where `K` is the number of parameters estimated in the regression. Using the empirical value of the test statistic and the degrees of freedom of the statistic, obtain the corresponding p value. In the code, the p value is denoted with `p_hu` where `hu` is shorthand for heteroskedasticity of unknown form.

`p_hu` is virtually zero. What do you conclude? What does this imply for using the usual or the robust standard error estimator of the OLS estimator? We reject the null hypothesis of homoskedasticity. Hence, we obtain statistical evidence that the error terms are subject to heteroskedasticity of an unspecified, or very general, form. The OLS estimator is no longer the most efficient estimator for this model although it is still unbiased and consistent. The usual standard error estimator will produce biased standard error estimates and therefore is not valid. We should use the robust standard error estimator to conduct hypothesis tests on population coefficients of the model.

Note that the number of the terms in the systematic component of the regression equation is increasing quickly with every additional new independent variables. This then increases the degrees of freedom of the Chi-squared distribution quickly. This means that the test is biased towards failing to reject the null of homoskedasticity because the Chi-squared distribution is flatter when its degrees of freedom parameter is larger.

```
u_hat_sq = u_hat.^2;
R = [dur ncb rank year];
C = [dur.*ncb dur.*rank dur.*year ncb.*rank ncb.*year rank.*year];
S = [dur.*dur ncb.*ncb rank.*rank year.*year];
X = [x_0 R C S];
LSS_hu = exercisefunctionlssrobust(u_hat_sq,X);
LM = N*LSS_hu.R2_c;
df = LSS_hu.K-1;
p_hu = chi2cdf(LM,df,'upper');
```

9. Test for heteroskedasticity of known form

A popular way of specifying the form of heteroskedasticity is exponential heteroskedasticity. That is, $u_i = e^{0.5X_i\gamma}\upsilon_i$ where it is assumed that $\upsilon_i \mid X_i \sim N(0,1)$ and X_i includes an intercept. Using $\mathrm{E}[u_i \mid X_i] = 0$, it is easy to verify that $\mathrm{Var}[u_i \mid X_i] = \mathrm{E}[u_i^2 \mid X_i] = e^{X_i\gamma}$. This shows that Ω depends on the k dimensional parameter vector γ. This conditional expectation function calls for a regression of u_i^2 on $e^{X_i\gamma}$. This regression is nonlinear in the coefficients, and estimation is difficult. Instead, we can first linearise u_i^2 by taking the logarithm of it, and then by taking the conditional expectation of $\ln(u_i^2)$ to obtain $\mathrm{E}[\ln(u_i^2) \mid X_i] = X_i\gamma + \mathrm{E}[\ln(\upsilon_i^2)]$. Here $\mathrm{E}[\ln(\upsilon_i^2)]$ is some constant and it does not depend on X_i. This conditional expectation function calls for a regression of $\ln(u_i^2)$ on X_i. u_i is not observed but it can be estimated with the OLS residual $\hat{u}_i$. In the regression of $\ln(\hat{u}_i^2)$ on X_i, joint significance of the coefficients provides statistical evidence that the errors of the assumed model are heteroskedastic in terms of the given form. The OLS estimator gives consistent estimates of all slope coefficients in γ. However, the intercept coefficient in γ, γ_0, and $\mathrm{E}[\ln(\upsilon_i^2)]$ make up the constant term of the regression equation, and therefore it can be shown that the OLS estimator of γ_0 is not consistent. However, this does not matter since it is good enough to approximate Ω up to some scalar not depending on X_i.

Carry out the test using the code presented at the end of the section. Create the dependent variable `u_hat_sq_log`. Create the systematic component of the regression. Use the supplied function to carry out the OLS regression. Use the R squared from this regression to construct the Lagrange multiplier test statistic. Using the empirical value of the test statistic and the degrees of freedom of the statistic, obtain the corresponding p value. In the code, the p value is denoted with `p_hk` where `hk` is shorthand for heteroskedasticity of known form.

p_hk is virtually zero. What do you conclude? We reject the null hypothesis that heteroskedasticity of the exponential form does not exist. We could claim the errors are subject to heteroskedasticity which could be explained by an exponential function. Obviously this does not say anything about the true form of the heteroskedasticity. It might be that another functional form could do a better job in explaining the conditional variance.

```
u_hat_sq_log = log(u_hat.^2);
X = [x_0 dur ncb rank year];
LSS_hk = exercisefunctionlssrobust(u_hat_sq_log,X);
LM = N*LSS_hk.R2_c;
df = LSS_hk.K-1;
p_hk = chi2cdf(LM,df,'upper');
```

10. Obtain the estimated transformation matrix

Assume that heteroskedasticity takes the exponential form as in the preceding section so that $\mathrm{Var}\left[u_i \mid X_i\right] = e^{X_i\gamma} = \Omega$. Therefore the model of interest is the generalised linear model. Estimate the coefficients of this model using the FGLS estimator. To carry out this estimation first obtain $\hat{\Omega}$. Then calculate $\hat{\Omega}^{-1}$. This is a diagonal matrix that contains the weights on the diagonal. It is a $N \times N$ matrix. Use the Cholesky decomposition $\hat{\Omega}^{-1} = \hat{\Psi}'\hat{\Psi}$ to obtain $\hat{\Psi}$. Transform the dependent and the independent variables by multiplying then with $\hat{\Psi}$. This in fact is just weighing, by dividing, each observation of the dependent and the independent variables with $\hat{\Psi}$ so that the residuals are weighed. Consequently, the residuals will have a variance corrected for the heteroskedastic structure it involves. Finally, apply the OLS estimator on the transformed variables using the supplied function file. Compare the FGLS estimates you obtain with the OLS estimates you have obtained in Section 5. What do you conclude?

Consider the code presented at the end of the section. The predictions $X_i\hat{\gamma}$ are given by LSS_hk.y_hat. Take the exponential function of these predictions with exp(LSS_hk.y_hat). It is a column vector. exp(LSS_hk.y_hat).*eye(N) places each element of exp(LSS_hk.y_hat) to the diagonal of a diagonal matrix where non-diagonal elements are always zero. Using the chol function on OI_hat, obtain a lower triangular matrix, which we name as P_hat. Then OI_hat = P_hat'*P_hat must hold.

```
Omega_hat = exp(LSS_hk.y_hat).*eye(N);
Omega_inverse_hat = inv(Omega_hat);
Psi_hat = chol(Omega_inverse_hat,'lower');
```

11. Obtain the FGLS coefficient estimates

Transform y and X using Psi_hat to obtain y_t = P_hat*y and X_t = Psi_hat*X where t is shorthand for transformation. Use the transformed variables as input arguments for the exercisefunctionlssrobust function. LSS_fgls contains the output this function returns. Call the FGLS estimates using the syntax B_hat_fgls = LSS_fgls.B_hat. Note how we use exercisefunctionlssrobust, which produces OLS coefficient estimates, to produce FGLS estimates. This shows that FGLS estimation is nothing but OLS estimation except that the observations of the variables are transformed. Compare the formulas of the two estimators to make this more explicit: the OLS estimator is given by `inv(X'*X)*(X'*y)` while the FGLS estimator is given by `inv(X_t'*X_t)*(X_t'*y_t)`.

The FGLS estimates are somewhat different from the OLS estimates. If the errors were homoskedastic, we would expect the weighting matrix `OI_hat` to matter less, and the FGLS and OLS estimates to be closer to each other. If we believe that the assumed form of heteroskedasticity is correct and the sample size is large, we would consider the FGLS estimates over the OLS estimates. Otherwise we can rely on the OLS estimates and for the standard error of these estimates we can use the heteroskedasticity-consistent standard error estimates.

```
y_t = Psi_hat*y;
X_t = Psi_hat*X;
LSS_fgls = exercisefunctionlssrobust(y_t,X_t);
B_hat_fgls = LSS_fgls.B_hat;
```

12. Obtain the standard error estimates of the FGLS coefficient estimates

The FGLS estimator is just the OLS estimator except that it uses `y_t` and `X_t` while the OLS estimator uses `y` and `X`. Therefore, the variance-covariance estimator of the FGLS estimator can be obtained directly using the standard variance-covariance estimator of the OLS estimator given in `exercisefunctionlssrobust.m` except that we need to replace `y` and `X` with their transformed versions `y_t` and `X_t`. The variance-covariance estimates are in fact already calculated in the preceding section and stored in the structure array `LSS_fgls.B_hat_VCE`. `LSS_fgls.B_hat_SEE` provides us with the standard error estimates of the FGLS coefficient estimates.

Compare the standard error estimates of the FGLS coefficient estimates with those of the OLS coefficient estimates. What do you conclude? The FGLS standard errors are slightly different from the OLS standard errors. This suggests that the errors are subject to heteroskedasticity. This means that the OLS estimator is not best anymore and therefore the FGLS estimator should be used. The FGLS standard errors appear to be always smaller than the robust OLS standard errors. This confirms that the FGLS estimator is efficient.

```
B_hat_SEE_fgls = LSS_fgls.B_hat_SEE;
```

§ Humor

> “The word econometrics should not be confused with economystics or economictricks.” Peter Kennedy, 2008. A Guide to Econometrics.