

## Empirical exercise – Function – Two-stage least squares statistics – Robust

This function calculates the TSLS coefficient estimates and related statistics. Note that for inference on the TSLS estimates, the  $z$  statistic is considered instead of the  $t$  statistic because the rationale for the TSLS estimation is asymptotic.

---

```
function LSS = exercisefunctiontslsrobust(y,X,Z)
% Number of observations and column dimension of X
LSS.N          = length(y);
LSS.K          = size(X,2);
%% Stage one (_so) of TSLS
% Estimates, predictions, residuals
LSS.B_hat_so   = inv(Z'*Z)*(Z'*X);
LSS.X_hat_so   = Z*LSS.B_hat_so;
LSS.u_hat_so   = X(:,2)-LSS.X_hat_so(:,2);
% The variance-covariance estimator of the OLS estimator
LSS.B_hat_VCE_so = inv(Z'*Z)*Z'* ...
    (1/(LSS.N-LSS.K)*LSS.u_hat_so'*LSS.u_hat_so.*eye(LSS.N))*...
    Z*inv(Z'*Z);
LSS.B_hat_SEE_so = sqrt(diag(LSS.B_hat_VCE_so));
% The variance-covariance estimator of the OLS estimator that is robust
LSS.B_hat_VCE_robust_so = inv(Z'*Z)*Z'* ...
    (LSS.u_hat_so.*LSS.u_hat_so.*eye(LSS.N))* ...
    Z*inv(Z'*Z)* ...
    LSS.N/(LSS.N-LSS.K);
LSS.B_hat_SEE_robust_so = sqrt(diag(LSS.B_hat_VCE_robust_so));
% Inference
LSS.t_df       = LSS.N-LSS.K;
LSS.t_so       = LSS.B_hat_so(:,2)./LSS.B_hat_SEE_so;
LSS.p_so       = tcdf(abs(LSS.t_so),LSS.t_df,'upper')*2;
% Inference robust to heteroskedasticity
LSS.t_robust_so = LSS.B_hat_so(:,2)./LSS.B_hat_SEE_robust_so;
LSS.p_robust_so = tcdf(abs(LSS.t_robust_so),LSS.t_df,'upper')*2;
%% Stage two (_st) of TSLS
% Estimates, predictions, residuals
LSS.B_hat_st   = inv(LSS.X_hat_so'*LSS.X_hat_so)*(LSS.X_hat_so'*y);
LSS.y_hat_st   = X*LSS.B_hat_st;
LSS.u_hat_st   = y-LSS.y_hat_st;
% The asymptotic variance-covariance estimator of the OLS estimator
LSS.B_hat_VCE_st = inv(LSS.X_hat_so'*LSS.X_hat_so)*LSS.X_hat_so'* ...
    (1/(LSS.N-LSS.K)*LSS.u_hat_st'*LSS.u_hat_st.*eye(LSS.N))*...
    LSS.X_hat_so*inv(LSS.X_hat_so'*LSS.X_hat_so)* ...
    (LSS.N-LSS.K)/LSS.N;
LSS.B_hat_SEE_st = sqrt(diag(LSS.B_hat_VCE_st));
% The asymptotic variance-covariance estimator of the OLS estimator that is robust
LSS.B_hat_VCE_robust_st = inv(LSS.X_hat_so'*LSS.X_hat_so)*LSS.X_hat_so'* ...
```

```

(LSS.u_hat_st.*LSS.u_hat_st.*eye(LSS.N))* ...
LSS.X_hat_so*inv(LSS.X_hat_so'*LSS.X_hat_so)* ...
(LSS.N-LSS.K)/LSS.N* ...
LSS.N/(LSS.N-LSS.K);
LSS.B_hat_SEE_robust_st = sqrt(diag(LSS.B_hat_VCE_robust_st));
% Inference
LSS.z_st = LSS.B_hat_st./LSS.B_hat_SEE_st;
LSS.p_st = normcdf(abs(LSS.z_st),0,1,'upper')*2;
% Inference robust to heteroskedasticity
LSS.z_robust_st = LSS.B_hat_st./LSS.B_hat_SEE_robust_st;
LSS.p_robust_st = normcdf(abs(LSS.z_robust_st),0,1,'upper')*2;
end

```