# Lab01: Data Transformations, Bivariate Regression Analysis, Numerical Integration & Distributions

**Handed out:** Monday, February 1, 2021

**Return date:** Monday, February 15, 2021, at the eLearning link **Lab01Submit** in the **Lab01** folder.

**Grades:** This lab counts 13 % towards your final grade

**Objectives:** This lab focuses on the Box-Coxtransformation of individual variables and set of variables. The Box-Cox transformation and its reverse transformation in bivariate regression analysis. The calculation of the expectation and variance of distance decay functions. Finally, some distributional properties are evaluated.

**Format of answer:** Your answers (statistical figures and verbal description) should be submitted as ***hardcopy***. Add a running title with the following information: Lab01, your name and page numbers. You may use this document as template. Copy the requested statistical figures into your document. Trial and error answers will lead to a deduction of points. Label each answer properly with the bold task and sub-task headings. You are expected to hand in professionally formatted answers: use a fixed pitch font, like **Courier New**, for any  code the use mathematical type-setting when equations are required. Copy and paste figures into your document. Make sure that each figure has a proper ***caption*** describing its content.

## Task 1. Univariate Variable Exploration and Transformations [2 points]

Use the **CPS1985** dataset[[1]](#footnote-1) in the library **AER** to explore the distribution of the respondents’ **wage**.

**Task 1.1:** Find the best -value for the Box-Cox transformation (see **summary(car::powerTransform(*varName*))**). Could the *log*-transformation (i.e., ) instead of be used? Justify your answer. [0.5 points]

**Task 1.2:** For the untransformed (, optimal () and over-adjusted ( Box-Cox transformed **wage** variable repeat the following tasks and ***comparatively interpret*** the results:

[a] Draw properly constructed histograms of *all three distributions* and discuss their properties,

[b] evaluate their sknewness (see **e1071::skewness( )**), and

[c] test whether these variables are approximately normal distributed (see **ks-test( )** or **shapiro.test( )**).

Address also the questions: Which transformed variable *comes the closest* to the normal distribution? Is the transformation with over-compensating the inherent positive skewness in the **wage** variable? [1.5 points]

## Task 2: Explore the function powerTransform to achieve a multivariate normal distribution [1 point]

Read up in ’s online help about the function **car::powerTransfrom( )** in the **car** library. Use the variables **water81**, **water80**, and **water79** from the **Concord1.sav** data file.

**Task 2.1**: ***Simultaneously*** estimate the ***optimal set*** of Box-Cox transformation parameters for all variables so that the set of transformed variables becomes approximately multivariate normal distributed. Report your code to do the estimation. [0.5 points]

**Task 2.2:** Show the output and interpret the results. [0.5 points]

## Task 3: Confidence Intervals [2 points]

Continue with the **CPS1985** dataset for this task. To simplify things do not perform variable transformations.

**Task 3.1:** Run a bivariate regression of **wage** (dependent variable) on **education** (independent variable) and interpret the model estimates. [0.5 points]

**Task 3.2:** Calculate the 90 % confidence intervals around the estimated regression parameters. Can you draw the same conclusion as you did using the *t*-test in the **summary** output of task 3.1? [0.5 points]

**Task 3.3:** Scatterplot both variables and add the predicted regression line as well as the lower and upper 90% confidence interval lines around the ***point predictions***.(i.e., prediction interval in Hamilton and **interval="prediction"** in the **predict** function).

You may also want to enhance your plot by adding lines for the means of education and income as well as adding a title. [1 point]

## Task 4: Calibration and Prediction of a Bivariate Regression Model with Skewed Variables [4 points]

The **dBase** file **CampusCrime.dbf** has among other variables the count variables **crime** (number of crimes committed on university campuses) and **police** (size of the campuses police forces).

Note: To import **dBase** files use the  function **foreign::read.dbf( )**.

**Task 4.1:** Scatterplot **police** in dependence of **crime** including their box-plots along the margins. Is a data transformation advisable?

**Task 4.2:** Find a proper transformation of both variables in a way that the independent variable **crime** is approximately symmetrically distributed and that the transformation of the dependent variable **police** leads to approximately symmetrically distributed regression residuals.

**Task 4.3:** Test whether a ***log***-transformation (i.e., ) is appropriate for both, the dependent and the independent, variables.

Important: If the ***log***-transformation is appropriate ( is insignificant) then ***use it*** because their relationship can now be interpreted *in terms of elasticities*.

**Task 4.4:** Estimate the model in the transformed system and interpret the estimates. Also ***test*** if the elasticity (i.e., slope parameter) differs significantly from the neutral elasticity of 1, i.e., . This could be done manually by using ’s standard error from the regression output or by using the function **car::linearHypothesis**.

**Task 4.5:** Perform a prediction in the original data units and plot the ***median*** and ***expectation*** curves. Interpret the plot both in terms of the median and expectation curves.

**Task 4.6:** Provide a ***clean***  script that documents your analysis. Do ***not show*** the provided functions in the sample script.

## Task 5: Numerical Integration [2 points]

Evaluate the three distance decay functions along a line of around the central reference point zero.

over their full distance range . Show the ***clean*** code for the tasks below.

**Task 5.1:** Plot these three functions within a reasonable value range of the distance . How does the shape of the curves change with increasing power?

**Task 5.2:** Evaluate the areas , and underneath the three curves over their full support with ’s **integrate( )**-function.

**Task 5.3:** Calculate the expectation of the distances for all three distance decay functions . Could you predict the Expectation by just looking at the plot of the three curves?

**Task 5.4:** Calculate the variances of the distances for all three distance decay functions .

Hints: [a] The distance decay function must the rescaled by its area to make it a proper statistical density function, which integrates to one. [b] In  the professional integration function is **integrate( )**. Study the online help for this  functions. [c] To learn how to properly setup the functions to be integrated (aka, integrand), see for example the **RiemannSum.R** or the online help of the **integrate( )**-function. [d] The **integrate( )**-function is more robust and may accept as upper and lower bounds the parameter **Inf** and **-Inf** rather than an arbitrary truncation point. [e] Additional parameters, other than **x**, to the integrand function, such as or , can be passed in the **integrate( )**-function using its placeholder argument “**…**”. See the online help for the function **integrate( )**.

## Task 6: Hamilton Appendix 1 [2 points]

**Task 6.1:** Why is a *t*-distributed random variable with degrees of freedom when it is squared identically to the -distributed random variable with one degree of freedom in the numerator and degrees of freedom for the denominator? Hint: use the definition of both random variables. [0.8 points]

**Task 6.2:** Answer the questions **4 a**, **b**, **c**,and **d** in Hamilton’s exercises on page 300. [1.2 points]

Use Word’s equation editor to typeset your answers.

1. See pages 1-3 in Kleiber & Zeileis, 2008. Applied Econometrics with R. Springer Verlag. Available as *e-book* in UTD’s library. To import the data-frame use the statement **data(CPS1985, package=”AER”)** . [↑](#footnote-ref-1)