Random regression models

Luise A. Seeker

2017

# Introduction

This is a working example of a random regression analysis performed in the paper “Longitudinal changes in telomere length and associated genetic parameters in dairy cattle analysed using random regression models” (Seeker et al., 2018).

Random regression models have been used a lot in animal breeding for the analysis of longitudinal data (lactation curves!!). With them you can calculate average dynamics of the trait of your interest (fixed curve) for the whole tested population and individual trajectories for each animal/ human/ plant/ measurement unit in the study (random curve) as a deviation from that fixed curve. You can also calculate the change of genetic variance, permanent environment effects and heritability over the life of the animal/ human/ plant/ ~~measurement unit~~ and calculate how well repeated measurements correlate over time within the animal (let’s stick with “animal” for now and agree that humans are animals as well). You could also do a bi- or multivariate random regression analysis and estimate between-trait correlations over time. But we will start with something a little bit simpler.

In this specific case, we had previously detected no permanent environment effect and we are going to model a homogeneous residual variance across all age groups. If you haven’t understood a word of the last sentence, don’t worry. I just mentioned it, because I hope we are going to have some more examples soon that might be closer to the analysis you would like to do. We will start slowly and hopefully progress slowly. I personally found random regression models scary and frustrating, because they are documented very poorly… Let’s change that☺

I am going to work with ASReml (Gilmour, Gogel, Cullis, & Thompson, 2009), R and Excel. I will try to document my scripts as well as possible. If you have questions, contact me and I will try to help you ([seeker.luise@gmail.com](mailto:seeker.luise@gmail.com)).

Also, contact me please if you find mistakes and I’ll try to fix them. Ok, Let’s start!

# Finding a suitable animal model:

I assume that you have some experience with linear mixed models. If not, you will find helpful online recourses elsewhere to get up to speed (sorry, if I am disappointing you right at the beginning).

We started by analysing our dataset (“cow dataset”) using conventional animal models (see Seeker et al. 2018, Scientific Reports). I briefly describe our experiment so you can understand what we did. Ideally this will enable you to see parallels to your own study system which will make it possible to transfer our methods to your own data set. We had following longitudinal dataset: 1328 telomere length measurements of 308 cows (=308 animal IDs). Telomeres are the ends of chromosomes that shorten in vitro with increasing number of cell divisions (Harley, Futcher, & Greider, 1990). We wanted to investigate, how telomeres in vivo change over life and if animals varied in their amount of change. So, telomere length was our trait of interest and therefore our response variable. Response variables (and their residuals!) should be normal distributed. In our case we log-transformed telomere data to achieve normality. The animal ID was fitted as random effect to account for multiple measurements within the individual. Telomere length was measured by quantitative polymerase chain reaction (qPCR). We ran 56 samples in triplicates on each plate and qPCR 25 plates in total. Each plate had 8 different rows. In former experiments we found out that both qPCR plate and row contribute to the measurement error. Therefore, we included both as fixed effects in our models (plate as factor with 25 levels and row as a factor with 8 levels). We also included the year of birth (YOB) of the animal as a fixed effect (factor with seven levels) and the animal’s age at sampling as a covariate (which means as a continuous measure).

Age is important for the random regression models, because it is our time variable. With random regression models you usually analyse repeated measurements that were taken along a time trajectory (days, months, seconds, centuries, or in our case age in months). To my knowledge, the time variable has to be on a continuous scale. Therefore, although we had fitted age as a factor with two levels (younger and older than 1 month) in the conventional linear mixed models, we are going to use age in months (0-75) for the random regression analysis.

We used ASReml (Gilmour et al. 2009) for the analysis which allowed us to include pedigree information. The random effect of the animal ID was fitted with and without pedigree information to enable the decomposition of the variance that is due to the animal into an additive genetic and a permanent environment effect (animal effect that is not due to its genetic makeup). In our case the permanent environment effect was zero. Be careful here, if your trait is influenced by permanent environment effects: Especially your heritability estimates will need to be calculated differently from what we do for our data set.

# Random regression models

The random regression models differed from the conventional animal model in the following:

* We included an additional fixed effect to the model: the genetic group of the animal as factor with two levels (S for selected for high milk yield and C for control). Although the genetic group did not affect telomere length significantly, we wanted to account for it in the genetic analysis of the trait.
* Age was fitted as a measure on a continuous scale (in months) not as a two-level-factor
* A polynomial was added to the fixed effect of age in months and to the random effect of animal ID to model non-linear changes of TL along the time trajectory
* (The animal effect without pedigree information was removed, because the environment effect was zero.)

Have a look at the working directory1 in which you will find lots of ASReml files, a pedigree file (PED\_FULL\_20160302.csv) and a data file (Stage1\_DF\_20170307\_REDUCED.csv). I use ASReml in Windows (LINUX users, don’t laugh!) with the editor “ConTEXT” (<http://www.contexteditor.org/index.php>).

Open the .as file (Stage1\_20170621\_RR\_tutorial.as). This is the file in which you write your models. I usually write several models in the same .as file and use the !DOPART in the first line to tell the program which model it should run.

I started by fitting the model from my last paper and to take it from there step by step. With the actual random regression models, I start in !PART 4 -7 where I fit polynomial functions of increasing order to the fixed effect of age in months. I use legendre polynomials because most people prefer them (yes, that’s the only reason and I am not sure WHY they are “better” then pol()… One of my PhD supervisors prefers pol() but also cannot exactly explain WHY…). Ok, back to my models. I run every model (specify which one you would like to run in !PART x in the first row and specify a file name extension after !ARGS and hit F9 if you use conTEXT). If you go back to the home folder you will find a .csv file called “01\_ModelComparison.csv”. In this one I copy the Akaike Information Criterion (AIC), and the LogL of each model. I use the AIC values for model comparisons (which is done all the time in evolutionary ecology and not at all in animal genetics. So beware of what is typically done for model comparisons in your field and listen to your supervisor/PI/wife/mum/…). Well, I looked at the AIC values and found them decreasing until I fitted a cubic function and increasing again for the quartic function. A smaller AIC value is favourable and a delta AIC value of 2 roughly correlates to a significance threshold of p=0.05.

At that point you could probably decide to plot only the fixed curve for the best fitting polynomial function to visualise how your trait changes with time across your study population. I calculated all four fixed curves using the Excel file “03\_CalcFixedCurve.xlsx”. However, before you can use it, you need to calculate an overall mean. For this, you can use my R scrip “02\_calculateOverallMean.R”.

# Calculation of the overall mean

For the calculation of the overall mean you need to generate a data frame that contains the number of measurements for each model term factor level. You can find an example for such a data frame in workingDirectory2 under the beautiful name “20170606\_NuOfMeasPerFixedEffectLevel.csv” (which stands for “number of measurements per fixed effect level” and describes exactly what it is). You list every factor level and count the measurements for this level. For example for the factor “plate” and the level “1” we had 56 measurements. You need to divide this number by the total number of measurements (1328) to get the relative number in column “Relative N”. Do that for every one of your factor levels and “TADAAA” you’ll have your own beautiful “NuOfMeasPerFixedEffectLevel file”.

Table : Example for a “relative number data frame”: data frame   
containing the number of measurements (N) for each model term   
factor level and the relative number N/1328 wich is needed for   
weighing effect sizes

|  |  |  |  |
| --- | --- | --- | --- |
| ModelTerm | Level | N | perc |
| Plate | 1 | 56 | 0.042169 |
| Plate | 2 | 56 | 0.042169 |
| Plate | 3 | 56 | 0.042169 |
| Row | F | 168 | 0.126506 |
| Row | G | 169 | 0.127259 |
| Row | H | 150 | 0.112952 |
| BirthYear | 2008 | 127 | 0.095633 |
| BirthYear | 2009 | 459 | 0.345633 |
| BirthYear | 2010 | 399 | 0.300452 |
| BirthYear | 2011 | 257 | 0.193524 |
| BirthYear | 2012 | 75 | 0.056476 |
| BirthYear | 2013 | 8 | 0.006024 |
| BirthYear | 2014 | 3 | 0.002259 |
| GENETICGROUP | S | 582 | 0.438253 |
| GENETICGROUP | C | 746 | 0.561747 |
| mu | 1 | 1328 | 1 |

Open the R script “02\_calculateOverallMean.R”. You need to provide the file path to your two input files. 1) The relative number data frame (like “20170606\_NuOfMeasPerFixedEffectLevel.csv”) and 2) the ASReml output file without extensions (such as .res, .asr or .sln). What the R script does is the following: It opens the .sln file which contains the solutions (effect sizes). It reads in all lines and removes those with an effect size of 0. It merges the solutions with the relative number file and calculates a weighted effect for each factor level. The sum of all weighted factor levels is the overall mean.

Table 2 shows an example of what the R script does. Note that not all model term levels are shown in the table.

Table : merged table containing infromation from table 1, from the .sln file and a newly calculated weighted effect. The sum of all weighted effects equals the overall mean.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ModelTerm | Level | Effect | seEffect | N | RelativeN | weightedEff |
| Plate | 1 | -8.18E-04 | 1.05E-02 | 56 | 0.042169 | -3.45E-05 |
| Plate | 2 | 9.45E-03 | 1.04E-02 | 56 | 0.042169 | 0.000398 |
| Plate | 3 | 6.30E-03 | 1.04E-02 | 56 | 0.042169 | 0.000266 |
| Row | F | -2.40E-02 | 5.99E-03 | 168 | 0.126506 | -0.00304 |
| Row | G | -1.73E-02 | 6.07E-03 | 169 | 0.127259 | -0.0022 |
| Row | H | -3.54E-02 | 6.10E-03 | 150 | 0.112952 | -0.00399 |
| BirthYear | 2009 | 1.07E-02 | 9.88E-03 | 459 | 0.345633 | 0.003712 |
| BirthYear | 2010 | -1.00E-02 | 1.06E-02 | 399 | 0.300452 | -0.00301 |
| BirthYear | 2011 | -7.87E-03 | 1.13E-02 | 257 | 0.193524 | -0.00152 |
| BirthYear | 2012 | -9.64E-03 | 1.37E-02 | 75 | 0.056476 | -0.00054 |
| BirthYear | 2013 | 3.56E-02 | 2.67E-02 | 8 | 0.006024 | 0.000215 |
| BirthYear | 2014 | 1.87E-02 | 4.05E-02 | 3 | 0.002259 | 4.22E-05 |
| GENETICGROUP | S | -2.62E-03 | 1.18E-02 | 582 | 0.438253 | -0.00115 |
| mu | 1 | 3.27E-02 | 1.75E-02 | 1328 | 1 | 0.03272 |

**The sum of all weighted effect sizes is the overall mean.**

# Generation of the fixed curve

With the calculation of the overall mean we are one step closer to the calculation of the fixed curve which describes the dynamics of your trait of interest with time at a population level. For generating the fixed curve we also need the .res file which contains the residuals for each time unit and order of polynomial, and the .sln file which contains estimates of the effect sizes. Both are automatically generated by ASReml.

A little bit of mathematical background. Don’t worry, it’ll be easy: The fixed curve is calculated using a polynomial function:

f(x)=anxn+an-1xn-1+...+a2x2 + a1x + a0

Don’t get scared. If you work through it one step at a time, it’s easy to understand. The intercept (a0) is the overall mean that we just calculated (see above). The coefficients of the polynomial (a1-an) are the solutions of effect sizes that can be taken directly from the .sln file. You find them behind the polynomial functions of 0th to nth order of the fixed effect (your time variable; see Figure 1). X-xn in the formula above are taken from the .res file (see Figure 2).

To make it easier, I wrote a little R script called “03\_extractFromResFile.R” that you can use to extract the information from the .res file and save them as .csv file. The first part of the script extracts residuals of the fixed effect (needed for calculating the fixed curve), the second part of the random effect (which is needed for the calculation of the random curves; see below). The first and the second part only differ, if a polynomial of different order is fitted to the fixed and the random effect (what we in our experiment try to avoid as you will see later).

When you are using the R script “03\_extractFromResFile.R” to extract data for the calculation of the fixed curve, use it for the model in which you used the highest order of polynomials for the fixed effects that you wanted to test. In our case I am using it for a cubic function.

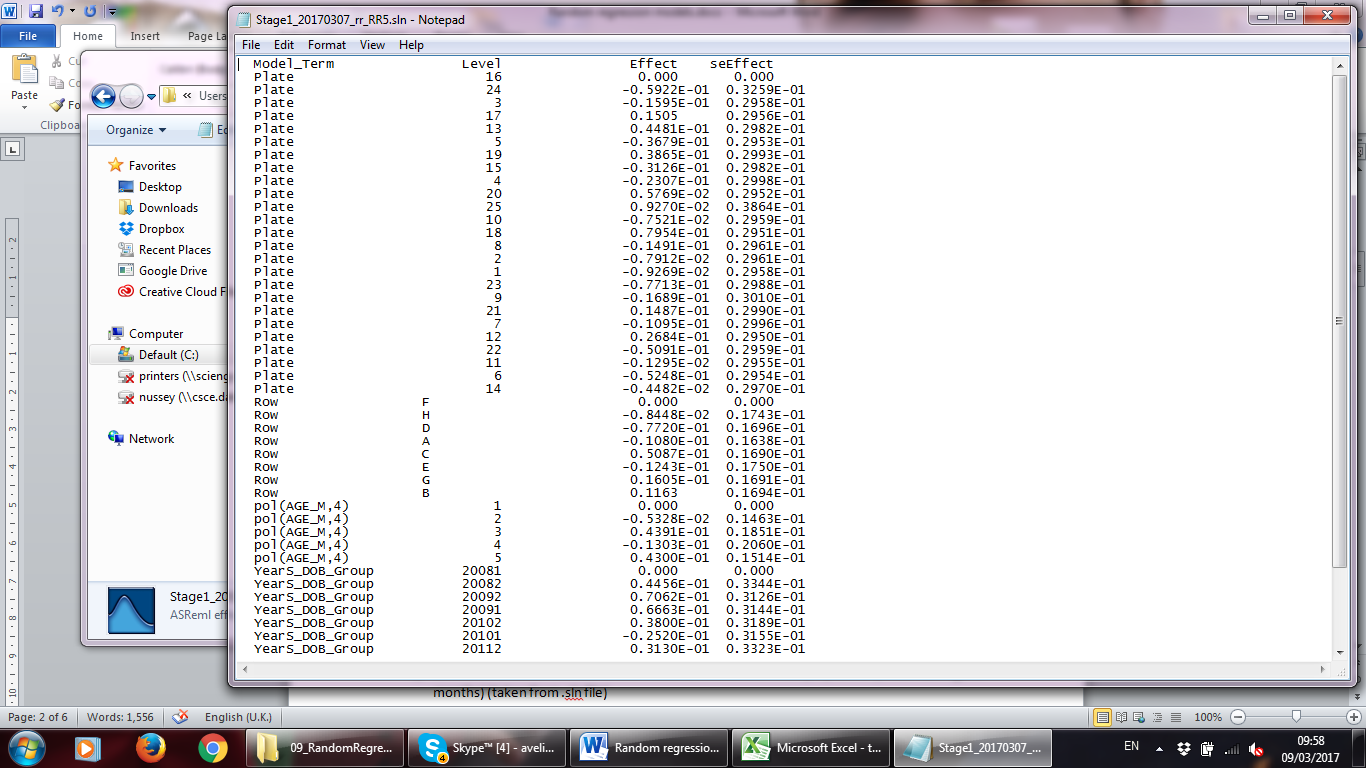


Figure : The .sln output file for a model containing a polynomial of 4th order for the fixed effect of age in months is shown. It demonstrates where the coefficients for the polynomial function come from. Level 2 is the estimated effect of the linear polynomial, level 3 is the effect of the quadratic, level 4 is the effect of the cubic and level 5 is the effect of the quartic polynomial. Level 1 represents a polynomial of the order 0 which is a parallel to the x- axis and therefore the effect size equals 0.

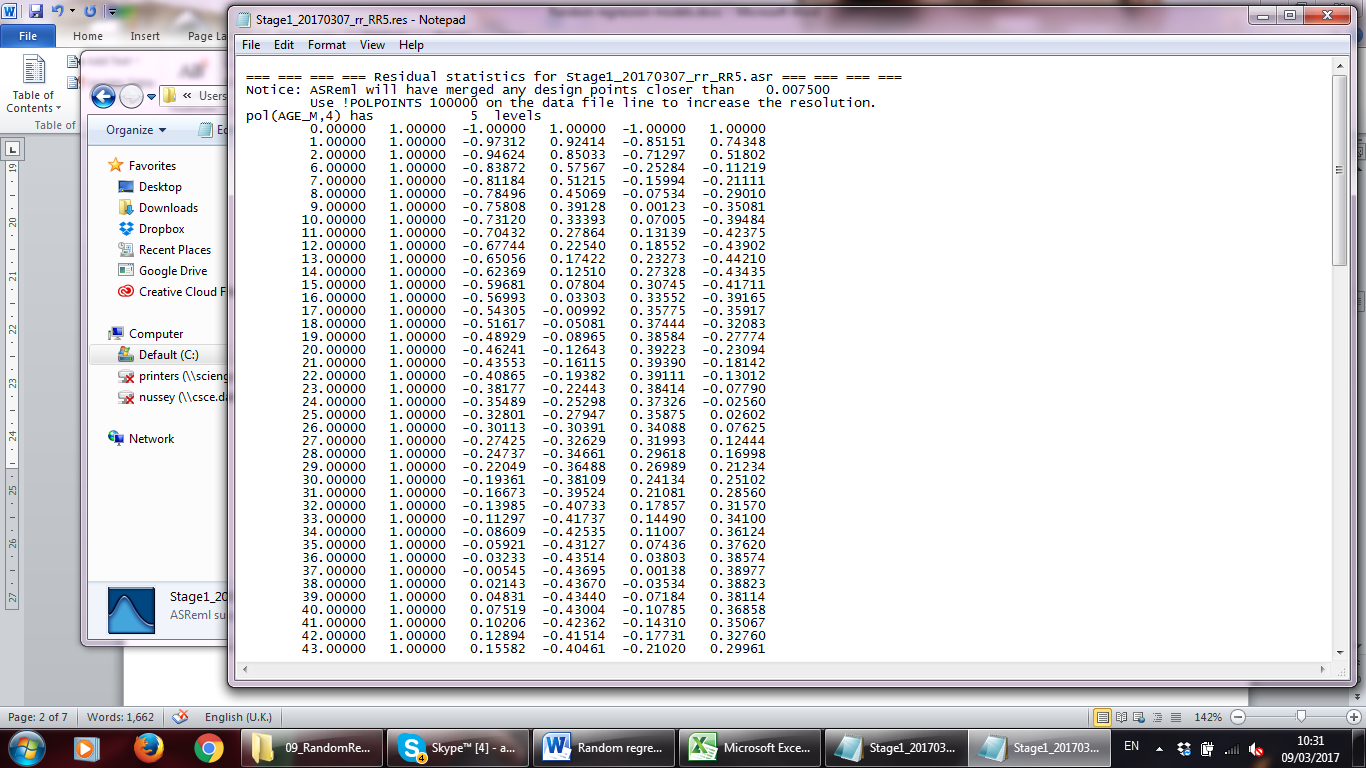


Figure : Example .res output of a model where a 4th order polynomial function was fitted to the fixed effect of age. First column: age in months, second column: 0 order polynomial, 3rd column: linear, 4th column: quadratic, 5th column: cubic, 6th column: quartic

To sum up, the following information is needed for calculating the fixed curve:

1. An overall mean (a0, see below for calculation)
2. Coefficients of the polynomial function (a1-an,taken from .sln file)
3. The residuals for each month and each order of polynomials (x-xn,taken from .res file)

Figure 3 shows the Excel template for the calculation of the fixed curve. The overall mean (mean) and data from the .sln and .res files are entered to the spread sheet and formulas for random regression functions are used for calculating estimates and in a second tap standard errors. The last tap I use to save measurement times (Age in months), the estimates of the polynomial of my choice (quadratic!), their standard errors and the phenotypic mean of the trait per time interval (in our case per month; calculated with the R script “05\_calculatePhenotypicMeanPerTimeInt.R”) . I save this tap as .csv file to use it as input file for the R script “06\_PlotFixedCurve.R”, because in my opinion plots generated with R are a lot prettier than those plotted with Excel.

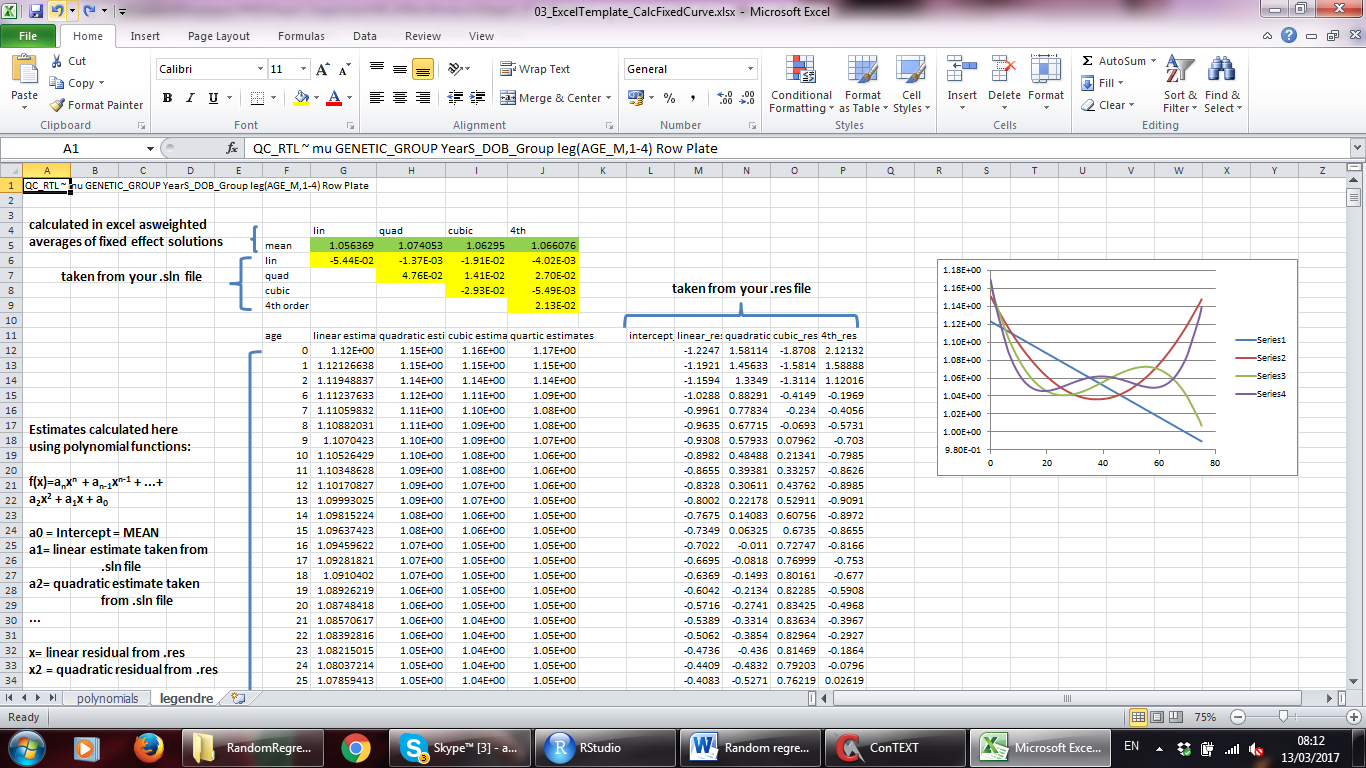


Figure : The Excel template for the calculation of the fixed curve is shown.

How do you decide which fixed curve fits your data best? You do so by optical evaluation, F-tests and comparison of AICs values. In our case the third order polynomial was the best fit. However, models with added polynomials for the random effect of animal ID did not converge with a third order polynomial for age in months. Therefore, we fitted the second best fit- a quadratic function- to the age term.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Source of variation | NumDF | DenDF\_con | F-inc | F-con | p-con | **AIC** |  |
| poly(age\_M,1) | 1 | 1076.7 | 76.68 | 75.31 | <0.001 | **-5907.18** |  |
| poly(age\_M,2) | 2 | 1064.5 | 73.56 | 75.55 | <0.001 | **-5963.41** |  |
| poly(age\_M,3) | 3 | 1056.9 | 58.27 | 59.54 | <0.001 | **-5975.00** | \* |
| poly(age\_M,4) | 1054.4 | 1054.4 | 44.65 | 46.67 | <0.001 | **-5970.12** |  |

# 5) Fitting a polynomial function to the random effect(s)

One exciting thing about random regression models is that you can plot a curve for each individual in your study. Each of them can have their own intercept and their own regression coefficient (therefore they are called “RANDOM REGRESSION” models!). So, how do you plot those random curves?

First you have to fit a polynomial function to your random effect which I found more complicated than doing the same for the fixed effect. Why is it more complicated? It’s complicated, because you have to model a variance structure in ASReml which is not really intuitive. I guess once you got it, you got it, but getting it is hard work…

Let’s start simple: We assume that the residual variance across all measurement times stays constant and that you have only one random effect (no permanent environment effect).

Ok, let’s go back to our .as file. You will see that once I found out that the cubic function fitted to the fixed effect fitted best and therefore I tried to fit the same order to the random effect (in model 8). Unfortunately this model did not converge. Therefore, I decided to fit a quadratic polynomial to both the random and the fixed effect (model 9). This makes the model a bit simpler and et voilà! The model converges.

Ok, let’s talk about modelling the variance structure. You might have noticed that there are a few lines in the .as file right beneath the model line. That’s what I am talking about.

The first thing you have to think about when modelling the variance structure is, if you want to assume that your residual variance is the same across all the time variables in your experiment (in our case age in months) or if you want to assume a difference variance structure for example for young and old animals. In jargon one would speak of a heterogeneous or homogeneous variance structure. In Fig. 1 you can compare two models, the first with a heterogeneous and the second one with a homogeneous variance structure. For the first it is crucial that your data file is sorted by your time variable (age in months). The three numbers following the model line (4 1 1) mean that four age groups are following and the last 1 means that 1 random effect is fitted. No one seems to know what the middle 1 stands for…

The next four lines are the numbers of samples within each time variable (age) group. They have to add up to you total number of samples. Sometimes you will see something like “0 0 !S2=10.” written behind those numbers which gives the model starting values for the residual variance. This is not necessary. The model runs without it.

Then you write your random effect term in the next line (what again does the 2 behind it mean?). US means “unstructured” and the following three values are the default starting values for your variance-covariance matrix. If you fit a quadratic function, you’ll need six starting values (but ASReml won’t crash if you give it again only three. It seems to be able to find starting values pretty well by itself). You can play around a bit with starting values. Sometimes it helps the model to converge or to converge faster. For this use the first sigma values that you can find in the corresponding .asr file).

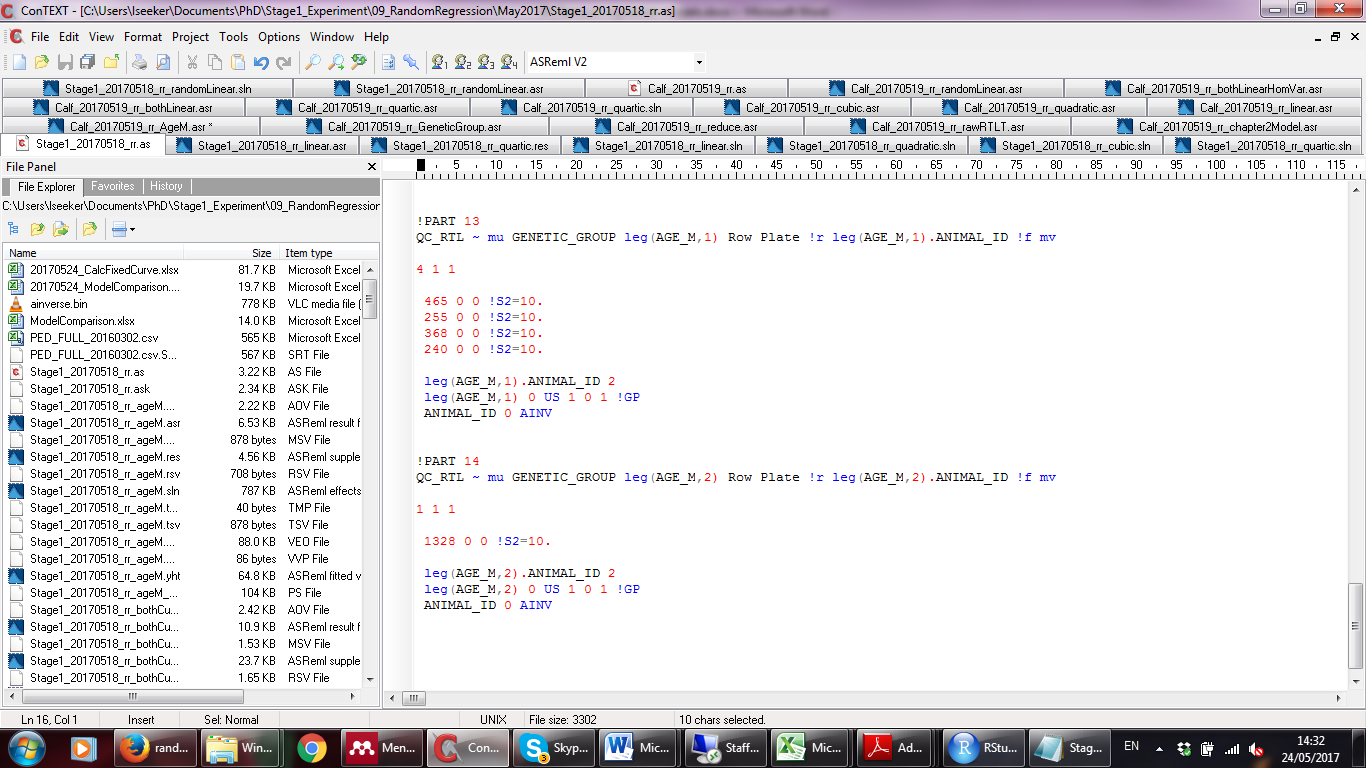


Figure : Modelling of the variance structure is shown for two different random regression models. The variance structure follows the model line. For the first model (!PART 13) a heterogeneous variance structure with four different age groups was modelled while for the second model (!PART 14) a homogeneous variance structure with all animals inside the same group was used.

The !GP keeps the variance-covariance matrix positive definitive. I had to remove this for my models to converge at least when I tested a heterogeneous residual variance.

When you add the random effect you can perform a log likelihood ratio test to test, if the polynomial fitted to the random effect improves the fit of the model. In our case it does which means that animals in our study vary in their telomere trajectories. If you should find out at that point that the random effect does not improve the fit, random regression models might not be the right way to analyse your data (I guess). If the model fit does not improve, trajectories might be truly very similar or you are lacking the power (too small sample size) to detect a variation that is actually there. You can also use delta AIC values to judge the fit of the models.

If possible, you can fit the permanent environment (ide(random effect)) as additional random effect. You should fit the same order polynomial to random effect and ide(random effect) and you have to add a variance structure for the permanent environment as well.

In our case the permanent environment effect was zero. Therefore, we did not add the ide(random effect) to our models.

# 6) Calculating the random curves

If you look at your solution file, you will find solutions for all the individuals in your model (and if you linked a pedigree file additionally of all individuals in your pedigree file). To restrict the creation of random curves to the animals for which you have measurements, you can re-run your model without the pedigree information or filter your .sln output for those animals in your experiment.

If you look more closely at the solutions for your individuals, you will notice that they have n+1 estimates, if n is the order of the polynomial you fitted. In our case with a polynomial of second order, we find three solutions in the .sln file for each animal. The individual ID that starts with a “1.” is associated with the intercept solution, the one that starts with a “2.” is associated with the linear solution, and the one that starts with a “3.” is associated with the quadratic solution (etc. if a polynomial of higher order is fitted) (see Figure 5).

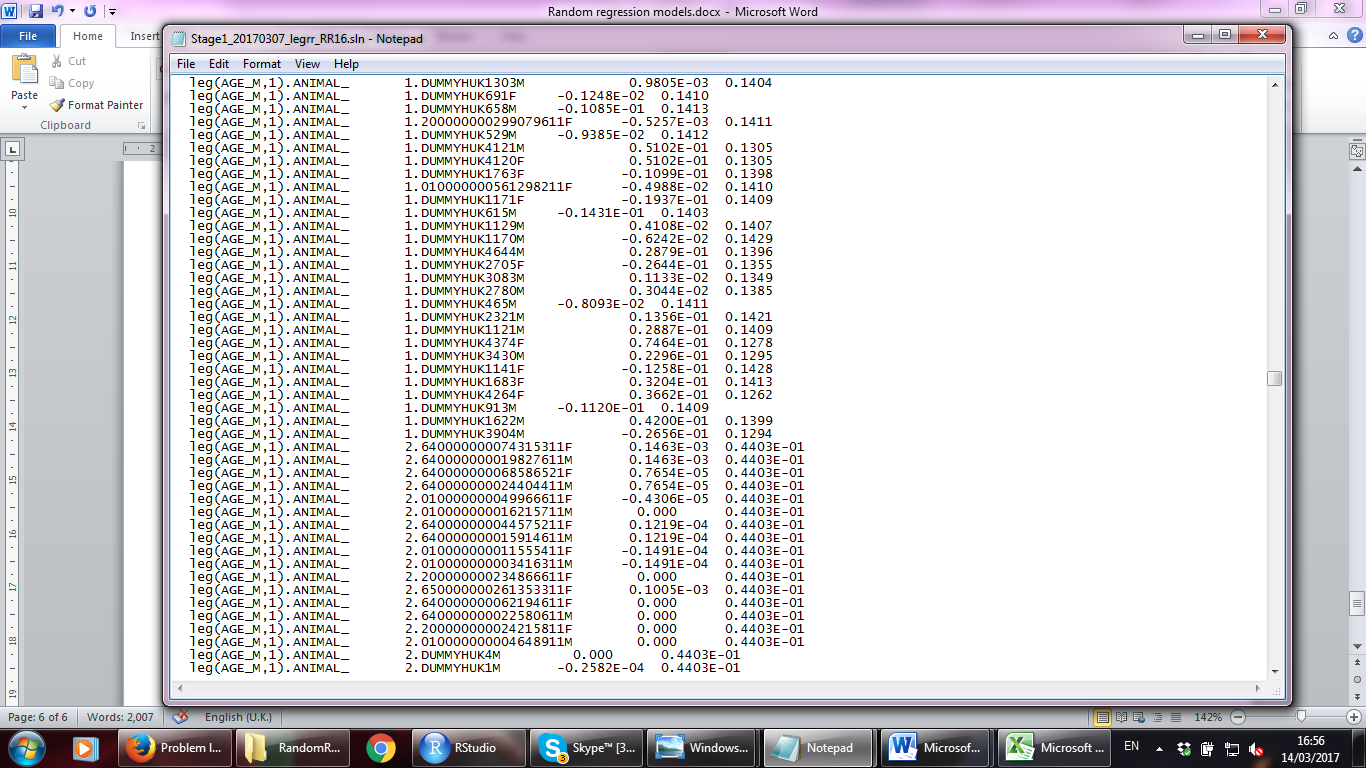
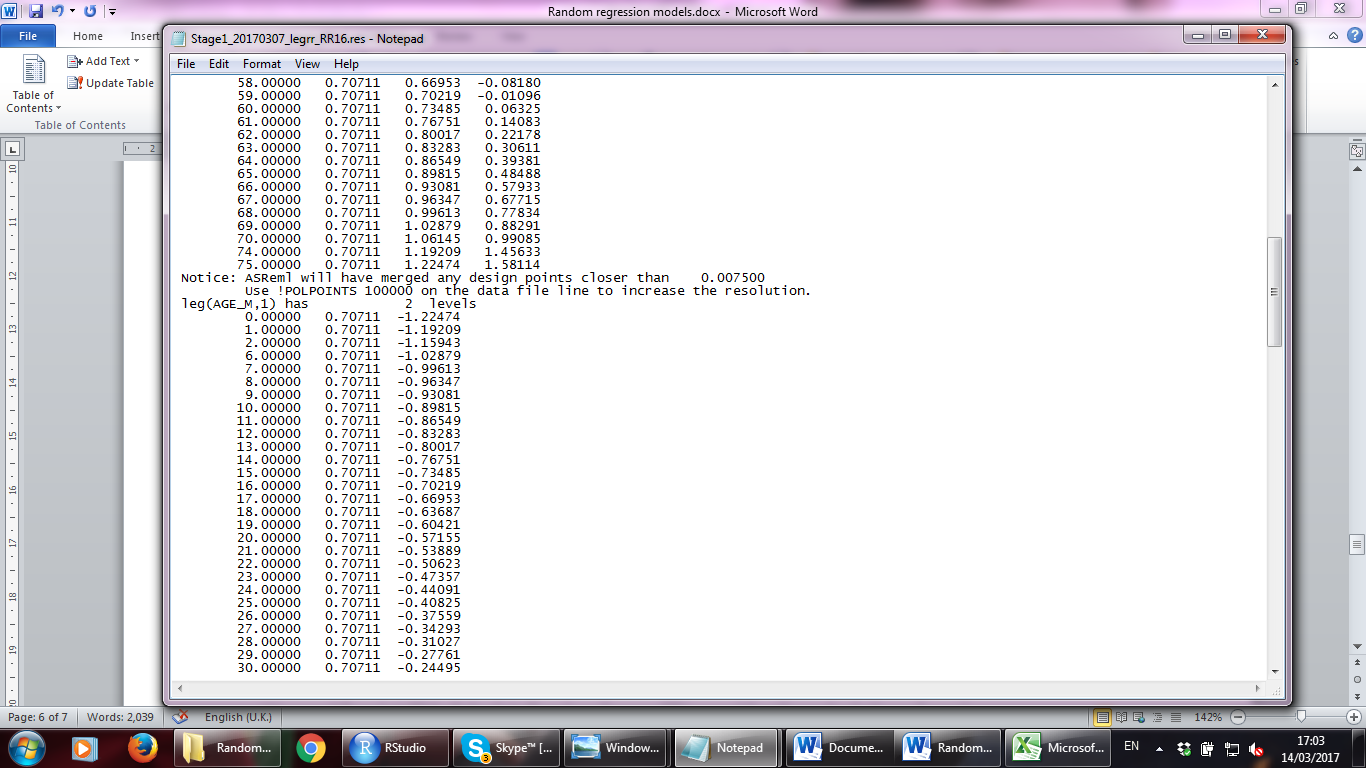
For calculating the random curves you have to multiply the intercept solution of each animal with the intercept residual for each time interval (month) taken from the .res file (see Figure 6). You do the same for the linear and the quadratic estimates (etc. if a polynomial of higher order was fitted) and add all products within animal and time interval (month). This will give you estimates for each time interval and each animal that you can use to plot individual trajectories.

Figure : example .sln file. Levels that start with "1." are solutions for the intercept, those that start with a "2." are solutions for the linear function, etc...

Intercept solutions

Linear solutions



Intercept residuals

Time points

Linear residuals

Figure : Example .res output. In the beginning of the .res file residuals of the fixed effect are shown. Right under them you will find the residuals of the random effect. The first column is for the time points for which measurements are present. The second column is for the intercept and the third for the linear residuals. The higher the order of your polynomial for the random effect, the more columns will appear here.

Ok, by this time you should have decided about the model you would like to use. I decided for model 9 in my .as file that contains a quadratic function fitted to both the random and the fixed effect and assumes a homogeneous residual variance across all ages.

At that point you should re-run the R script “03\_extractFromResFile.R” for the model of your choice, because you will need the output for the calculation of the random curve. There is an extra bit of the script that saves the residuals without the time interval and headers so you can use the output later directly as a matrix for the calculation of the random curve.

You will also need to extract information from your .sln file. For this you can use the R scrip “07\_ExtractFromSlnFile.R”.

We used the R script “04\_CalculateRandomCurves.R” for the calculation of random curves. The script reads in the modified previously created solutions and residuals data frames in, re-sorts the .sln file so that it has separate columns for the estimated effect of each order of the polynomial function. It adds for each animal as many rows as there are time points with measurements in the .res file. It also adds to this column the time points for which measurements are present. This column is used to merge the data frame with a residual data frame. Then, the intercept effect from the .sln file is multiplied by the intercept residual for each day and the same is done for linear and quadratic (etc. if polynomials of higher orders are fitted) solutions and residuals. The sum of these three (or more) products is the measure of interest which is used for plotting the change of the trait for each animal over the time trajectory.

The question is what you can do with those individual plots. You could use them to visualise that different animals follow different trajectories. Some people say you should never stick those results into the next analysis which often results in false positive results.

In the script for the calculation and plotting of random curves, I cluster them very roughly in two or three groups and use the cluster groups in a cox-proportional hazard model. This is exactly what some people say to never do. But I am doing it with the intention to show that random regression models are not the right analysis to understand within- individual change of telomere length.

But there are other cool things you can do with those models:

You can for example calculate within (and between) trait correlations over time and the heritability over time. Both I will show you below. Let’s start with the heritability calculation over time. You can use either the R script “09\_CalcHeritability.R” or the Excel template “10\_calculateSEforh2.xlsx” for the calculation of the heritability. The R script has the advantage that it saves your variance-covariance matrix as .csv file, the Excel template has the advantage that it also calculates the SE of heritability. If you use the Excel template, remember to press Ctrl+Shift+enter for executing the matrix calculations. Otherwise, you will get a #VALUE! error.