

Linear Mixed Model for Engelmann Spruce

Courtney Giebink

August 14, 2019

Data Summary

```
load(file = 'C:/Users/clgie/OneDrive/Documents/Masters/Utah/UT_FVS/data/formatted/data_all_es')

data_all_es <- data_all_es %>%
  mutate(dds = (2*RW*0.0393701)^2) #mm to inches
```

My data comes from the U. S. Forest Service's Interior West - Forest Inventory and Analysis (IW-FIA) program (DeRose, Shaw, and Long 2017). The U.S. Forest Service's Forest Inventory and Analysis program has established plots across the nation, which they use to collect data on forest stands. Recently, tree rings associated with plots in the Interior West states were discovered in storage. The IW-FIA program merged these two data sources: inventory data and tree-ring data. The data that I will use for my analysis is inventory and tree-ring data for *Picea engelmannii* (Engelmann Spruce) in Utah. I was able to extract and calculate annual data for 26 trees between the years of 1962 and 1994.

```
#create new dataframe with only variables needed for model
#response: RW, dds
#fixed: SI, ASP, SL, DBH/DIA_C, BAL, CR, CCF, PCCF, climate
#random: TRE_CN, Year
#filter for last 30 years of growth
min(data_all_es$MEASYEAR) #1992 -> 1962
```

```
## [1] 1992
```

```
glmm_data_es <- data_all_es %>%
  select(PLT_CN, TRE_CN, RW, dds, Year, DIA_C,
         SICOND, ASPECT, SLOPE, BAL, CR, CR_weib, PCCF, CCF,
         ppt_Jul, ppt_Apr,
         ppt_pJunAug, ppt_JunAug, ppt_pJulSep, ppt_pOctDec,
         ppt_pJunNov, wateryr,
         tmax_pAug, tmin_Feb, tmin_Mar,
         tmin_FebApr, tmax_pJulSep, tmin_JanMar,
         tmin_pNovApr) %>%
  filter(Year >= 1962)

#climate
#total ppt
#1 month: Jul, Apr
#3 month: pJun-pAug, Jun-Aug, pJul-pSep, pOct-pDec
#6 month + : pNov, wateryr

#average temp
#1 month: tmax_pAug, Feb, Mar
#3 month: tmin_Feb-Apr, tmax_pJul-pSep, tmin_Jan-Mar
#6 month + : tmin_Apr
```

My data, `glmm_data_es`, consists of raw radial increment, or ring width value, (RW) in mm and annual change in squared inside bark diameter (dds) in inches² with corresponding tree, climate, and site variables. Some variables stay constant, such as site index (`SICOND`), aspect (`ASPECT`), and slope (`SLOPE`), while diameter at breast height (`DIA_C`) and crown ratio (`CR`, `CR_weib`) vary annually. In addition, density variables, being basal area of trees larger than the subject tree (`BAL`), crown competition factor on the inventory point (`PCCF`), and stand crown competition factor (`CCF`), vary annually. Significant seasonal precipitation and temperature variables were identified based on tradition dendro analysis tools, using R packages `dplR` and `treeclim` (see `Climate-growth.Rmd`)

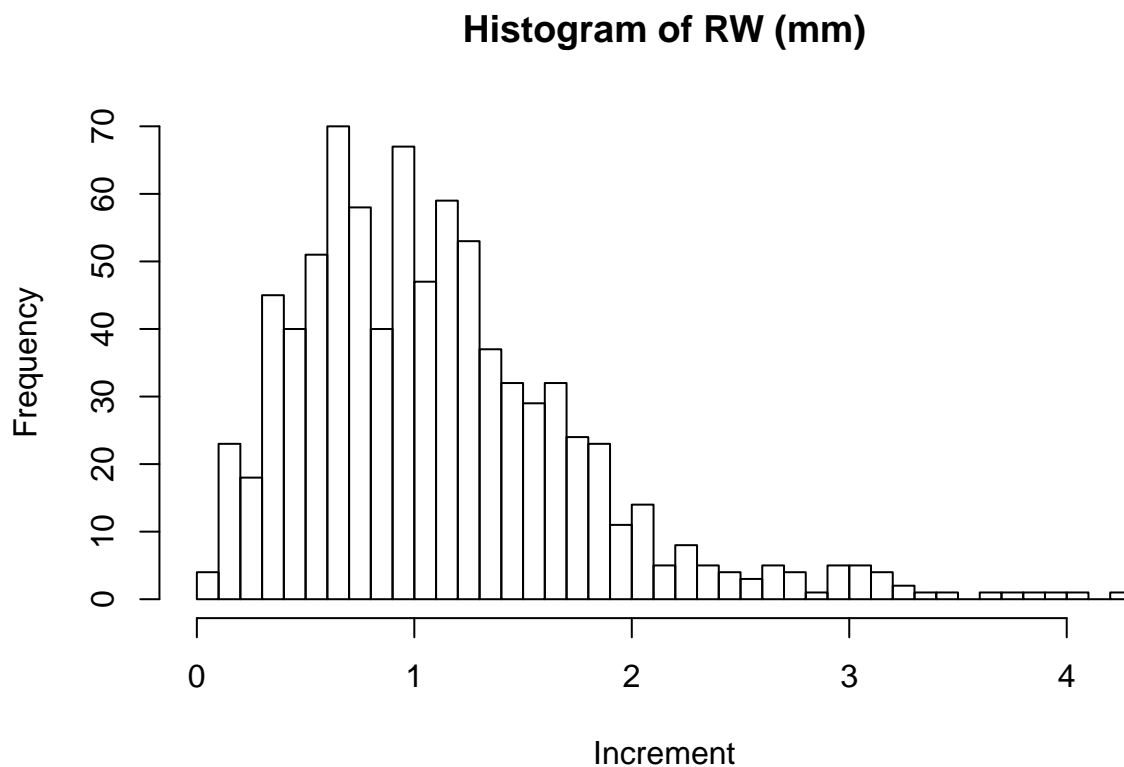
Exploration

It is necessary to explore the dataset to determine if the data complies with model assumptions (Zuur, Ieno, and Elphick 2010).

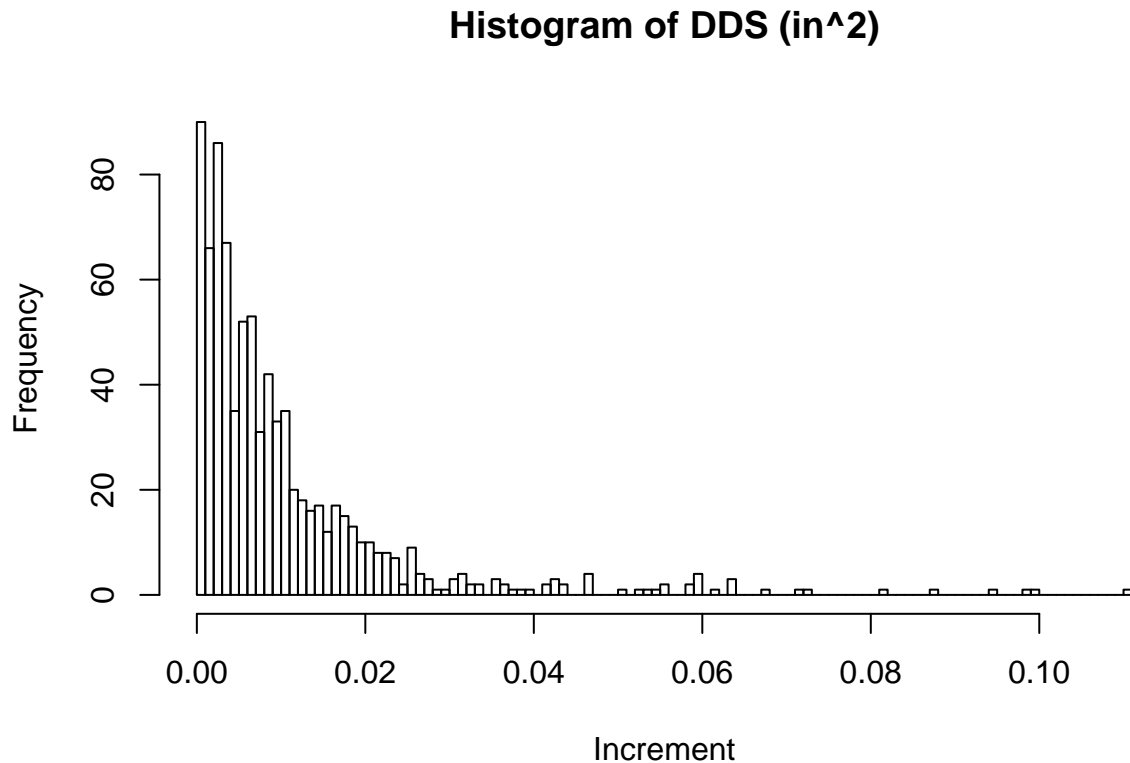
Normality

The response variable, annual change in squared inside bark diameter (DDS), is continuous. It will be explained using a linear mixed model.

```
hist(glmm_data_es$RW,breaks = 50,  
     main = "Histogram of RW (mm)", xlab = "Increment")
```



```
hist(glm_data_es$dds,breaks = 100,
     main = "Histogram of DDS (in^2)", xlab = "Increment")
```



The distribution of DDS is highly skewed right, while the distribution of raw radial increment (RW) is only slightly skewed right. The Gamma distribution is often used for skewed data, although Gaussian might still be appropriate and easier to interpret.

```
length(which(glm_data_es$RW == 0)) #1
```

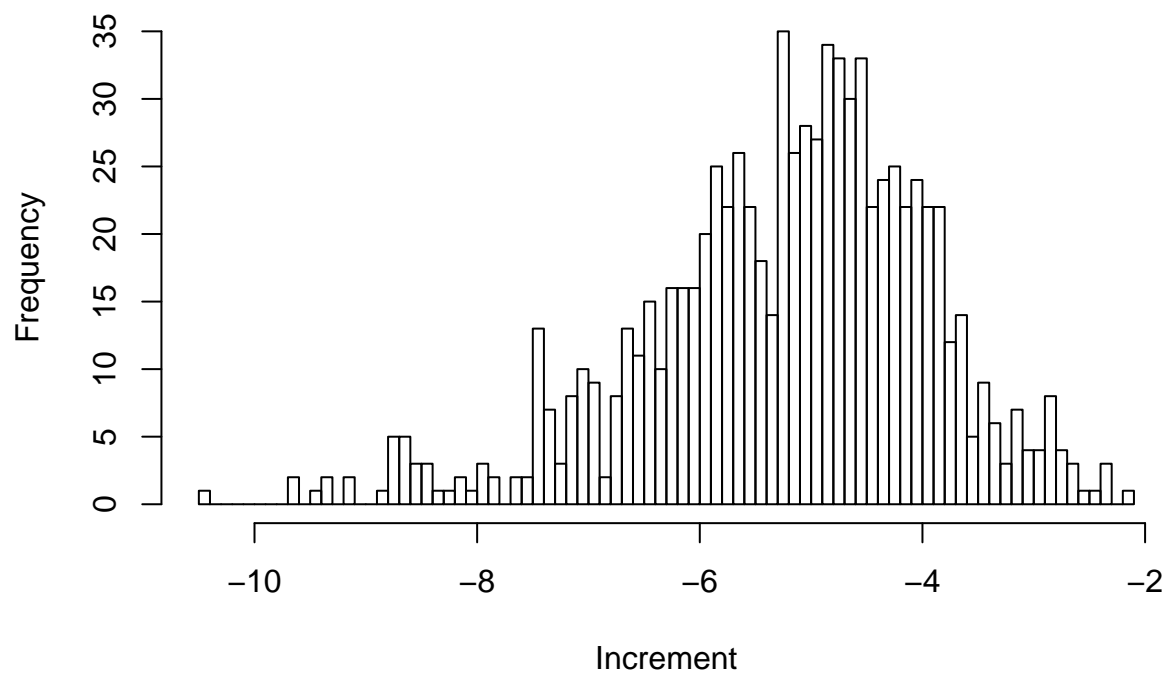
```
## [1] 1
```

```
#missing rings
which(is.na(glm_data_es$RW)) #0
```

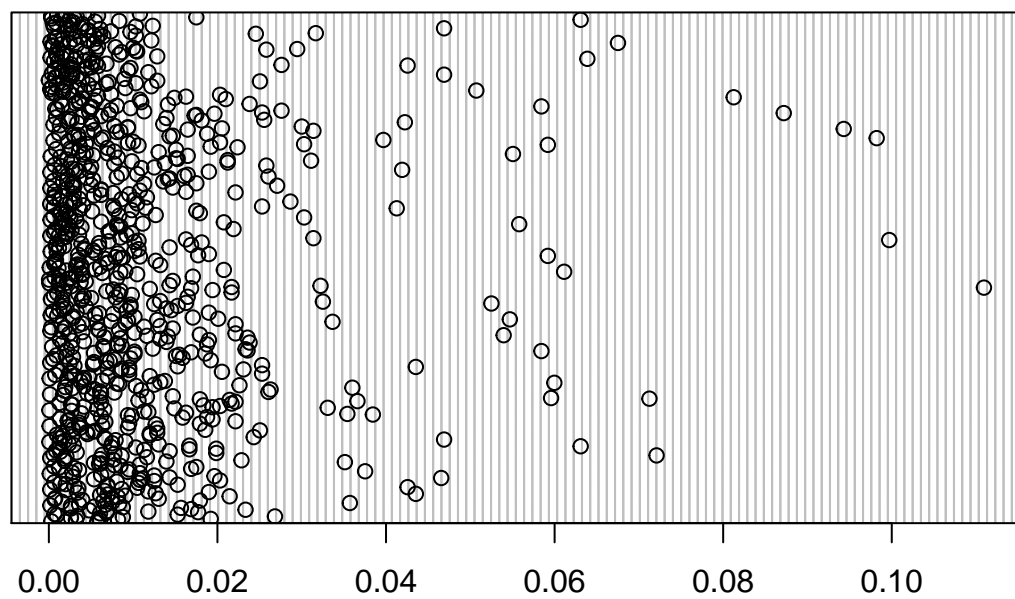
```
## integer(0)
```

```
hist(log(glm_data_es$dds),breaks = 100,
     main = "Histogram of log(DDS)", xlab = "Increment")
```

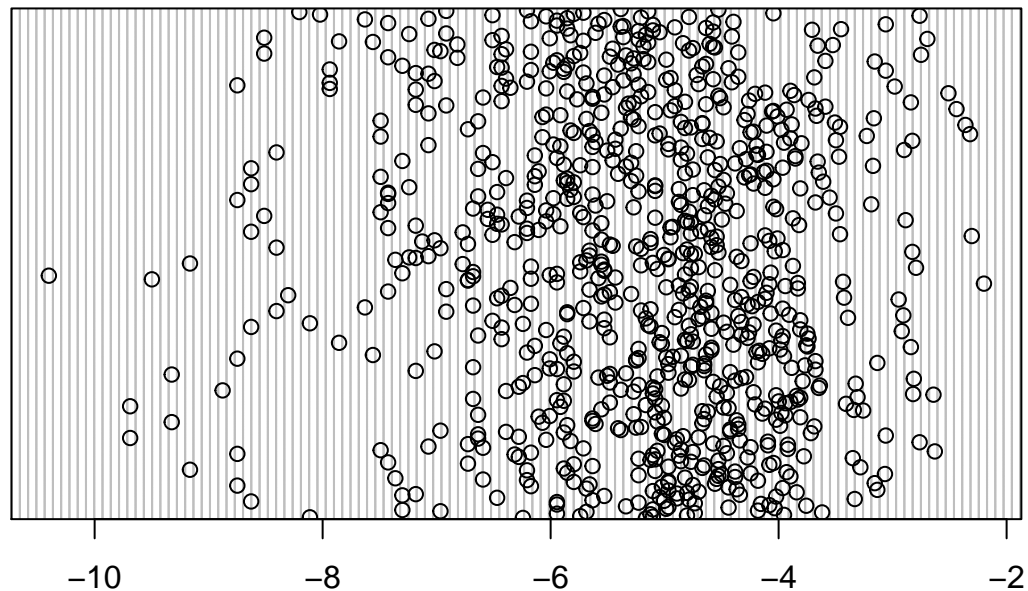
Histogram of log(DDS)



```
#Cleveland dotplot  
dotchart(glm_data_es$dds)
```



```
dotchart(log(glm_data_es$dds))
```



A log transformation normalizes the distribution of DDS and gets rid of a majority of the outliers.

Zeros and Missing Data

```
length(unique(glm_data_es$TRE_CN[glm_data_es$Year == 1962]))
```

```
## [1] 26
```

My data is unbalanced. All trees have a growth measurement for 1962, but the last year of growth is not the same for all trees. Some trees end growth in 1992, but some end in 1993 or 1994. There are no missing data in between first and last year of growth for each tree, and there is one year of zero growth for one tree, indicating a missing ring.

Random and Fixed Effects

In the linear mixed effects model, tree, stand, and climate variables will be fixed effects and a tree identifier and year of growth will be random effects.

```
length(unique(data_all_es$TRE_CN))
```

```
## [1] 26
```

```
length(unique(data_all_es$PLT_CN))
```

```
## [1] 23
```

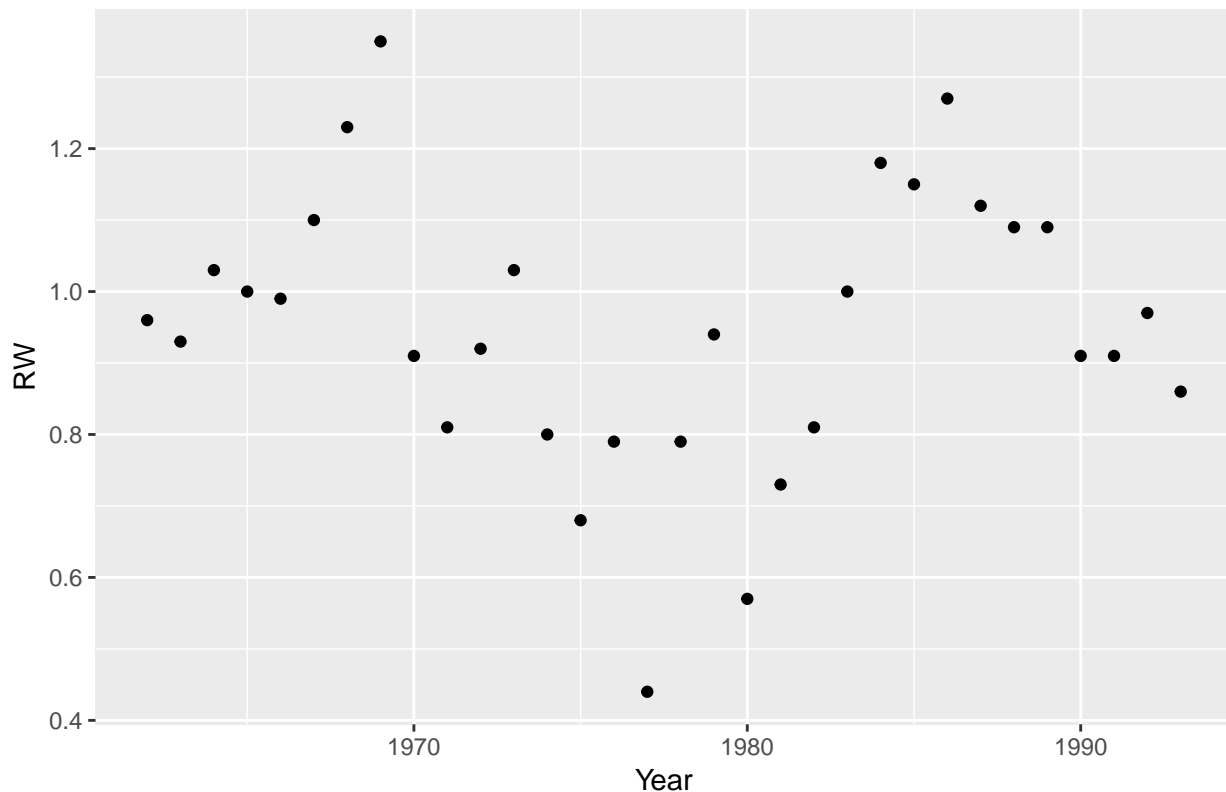
There are almost as many unique plots as unique trees, so it is not necessary to have plots as a random effect as well.

Tree ID and year are appropriate random effects because selected trees and years are a subset from all the possible selected trees and years (Harrison et al. 2018). Each tree has its own intercept of growth due to factors not considered here, such as genetics, microclimate, etc. Each year has its own intercept of growth due to relatively good or poor conditions not accounted for in temperature and precipitation, such as cloud cover, vapor pressure deficit, etc. Each tree and year might also have its own slope, but convergence on a random slope effect can only occur with an adequate amount of data.

```
#understanding random effects: TRE_CN
growth_yr_plots_es <- glmm_data_es %>%
  group_by(TRE_CN) %>%
  do(plots=ggplot(data=.) +
    aes(x=Year, y=RW) + geom_point() + ggtitle(unique(.$TRE_CN)))

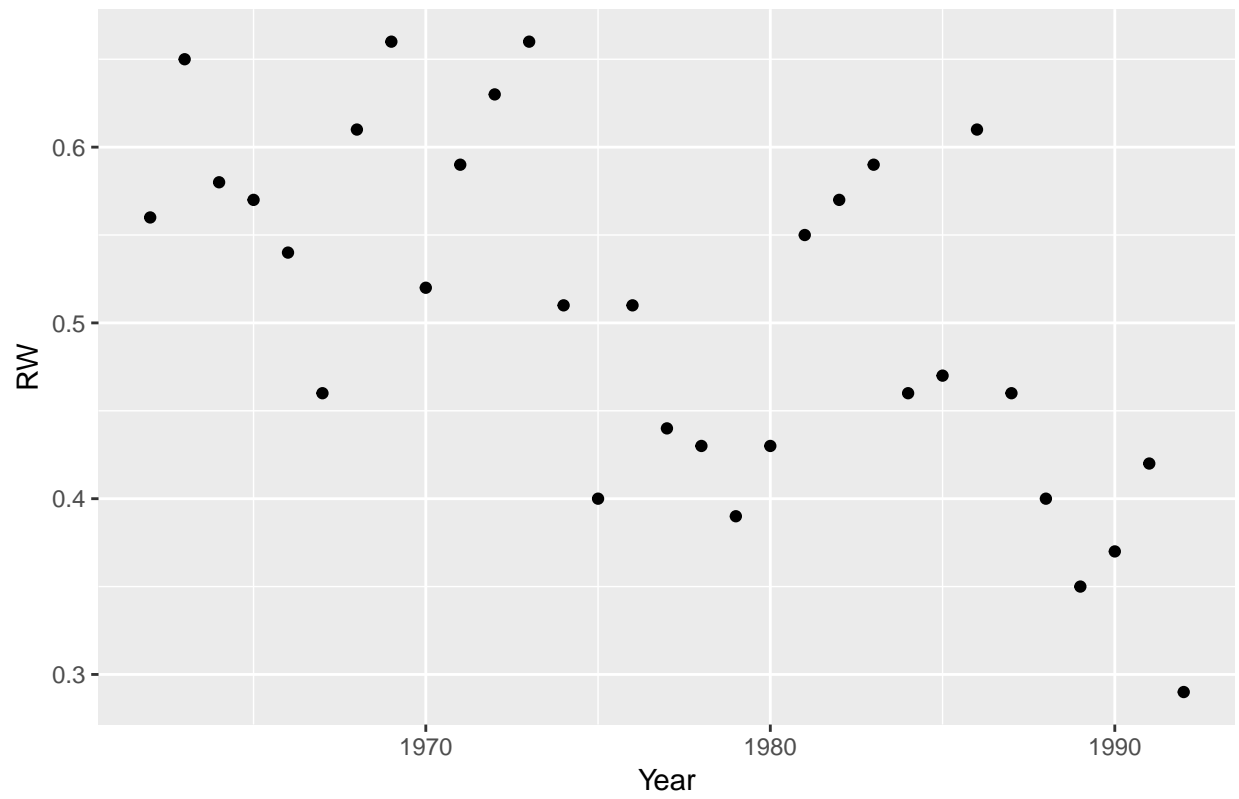
growth_yr_plots_es$plots[[1]]
```

2849671010690

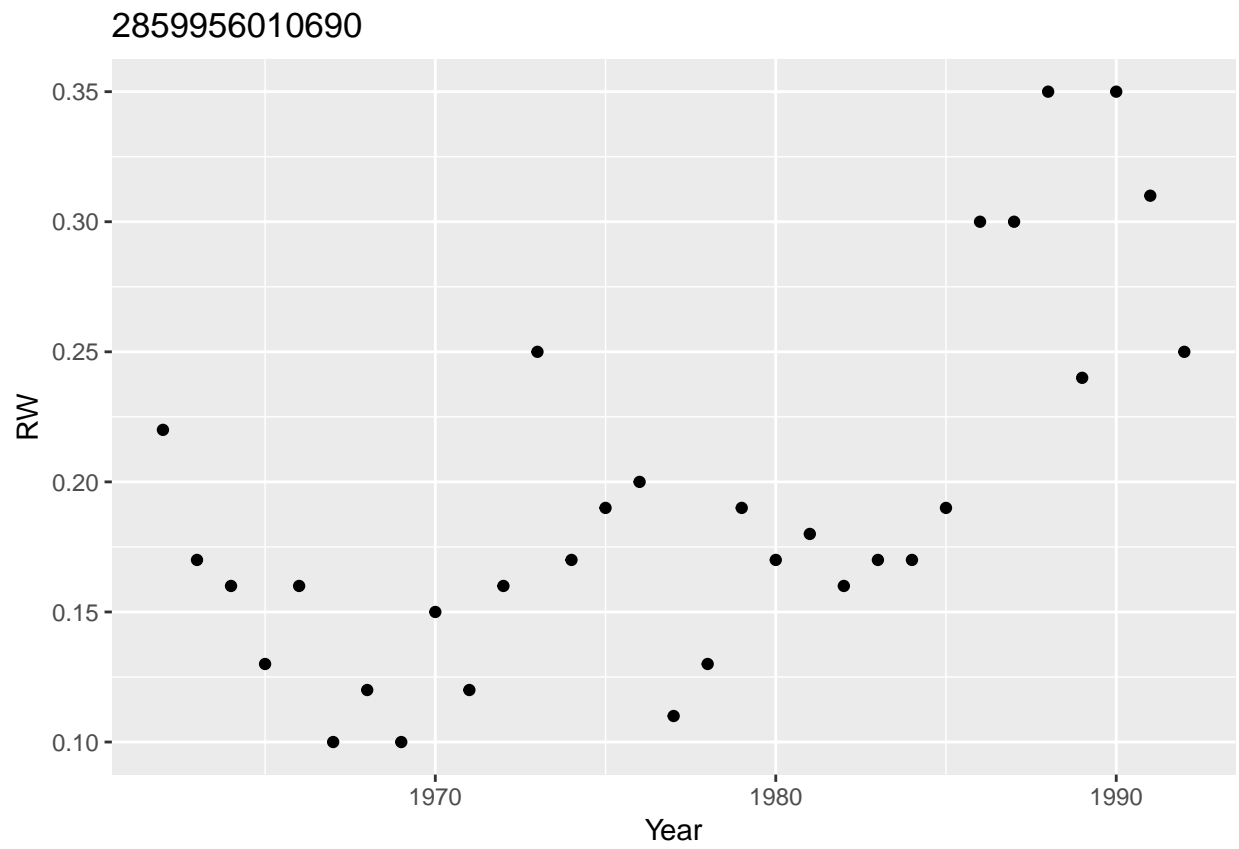


```
growth_yr_plots_es$plots[[2]]
```

2859936010690

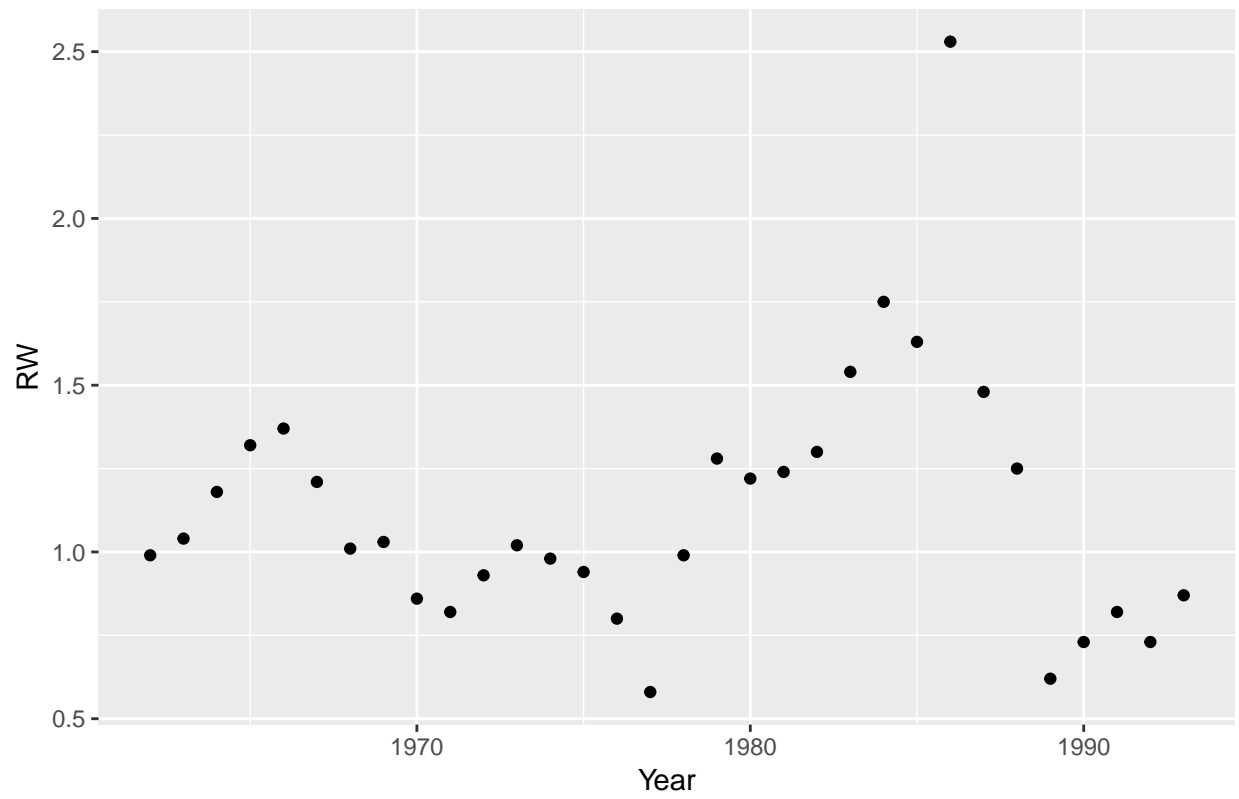


```
growth_yr_plots_es$plots[[3]]
```

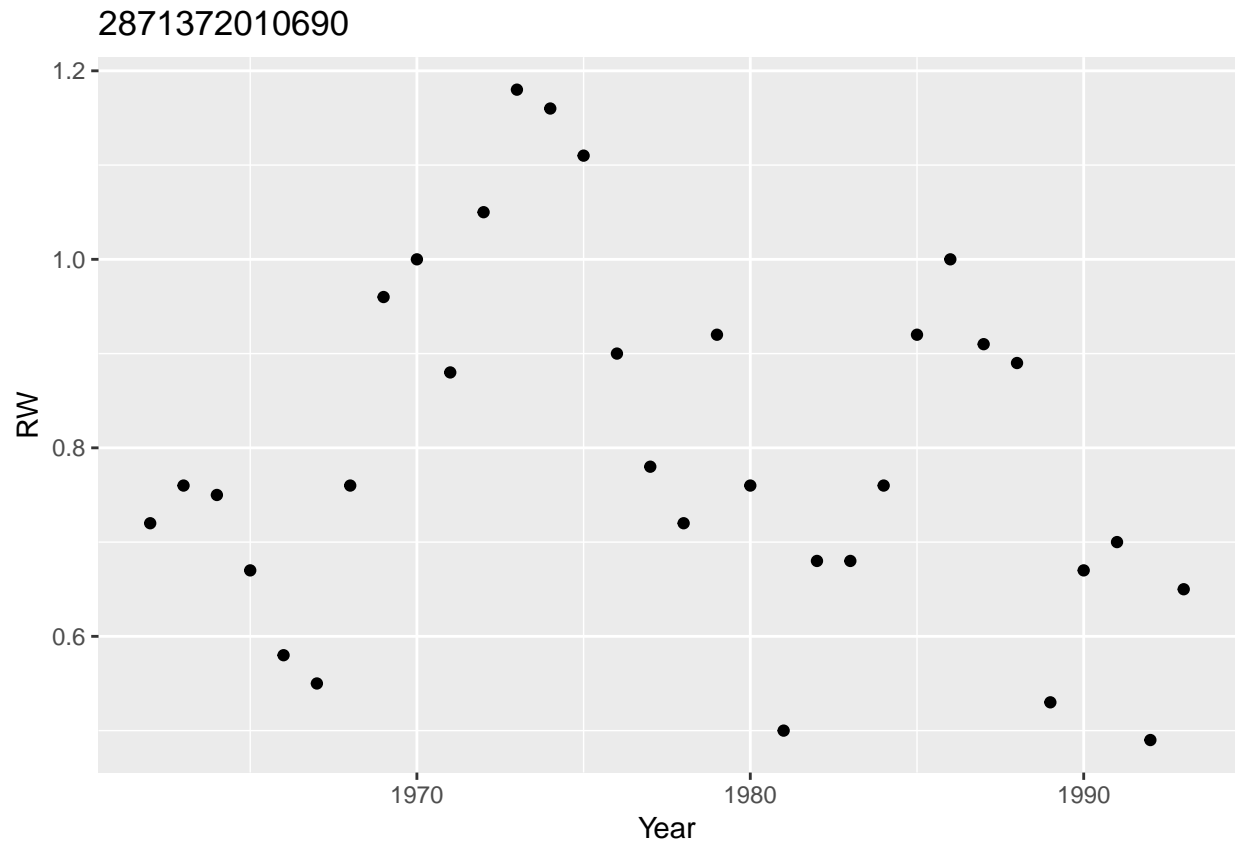



```
growth_yr_plots_es$plots[[4]]
```

2868676010690



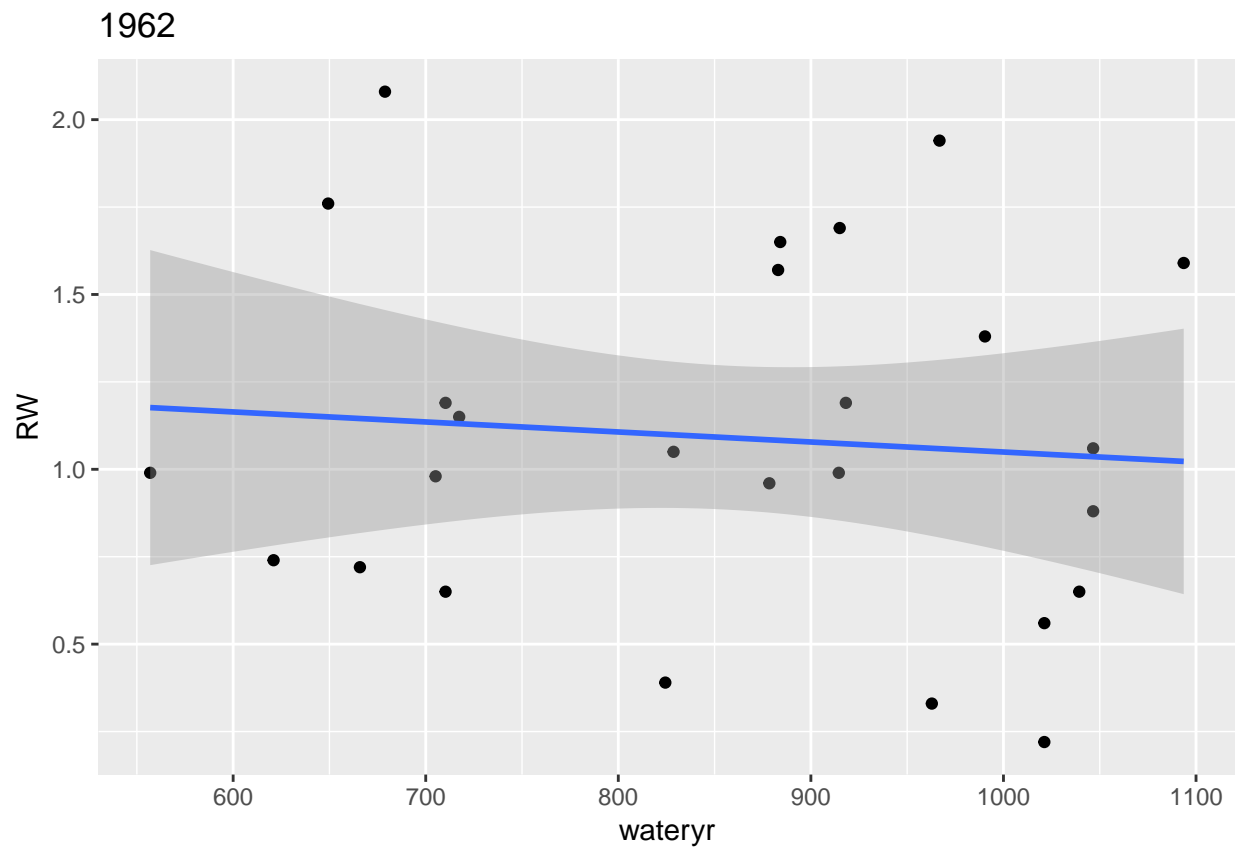
```
growth_yr_plots_es$plots[[5]]
```



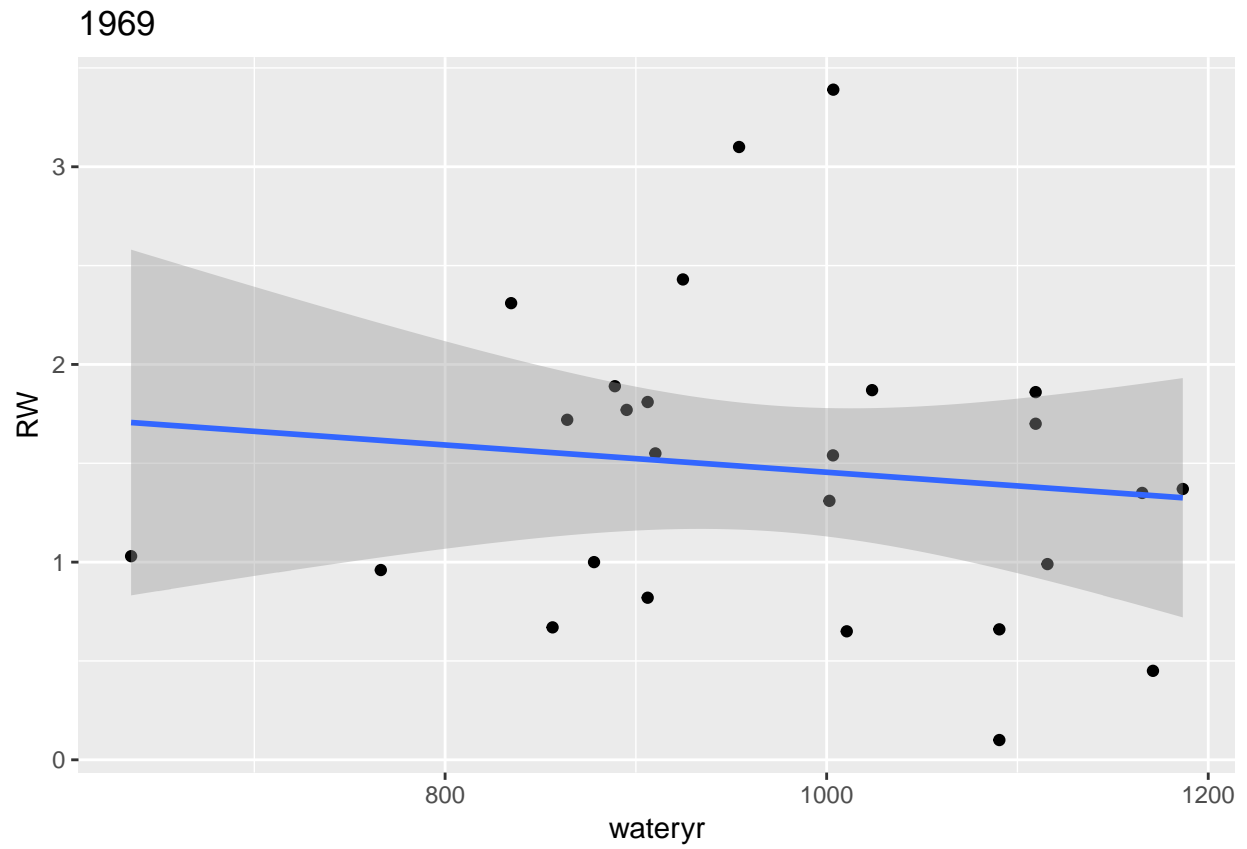
#can do more all the way to tree 26

```
#understanding random effects: Year
growth_wateryr_plots_es <- glmm_data_es %>%
  group_by(Year) %>%
  do(plots=ggplot(data=.) +
    aes(x=wateryr, y=RW) + geom_point() +
    geom_smooth(method = "lm") + ggtitle(unique(.$Year)))

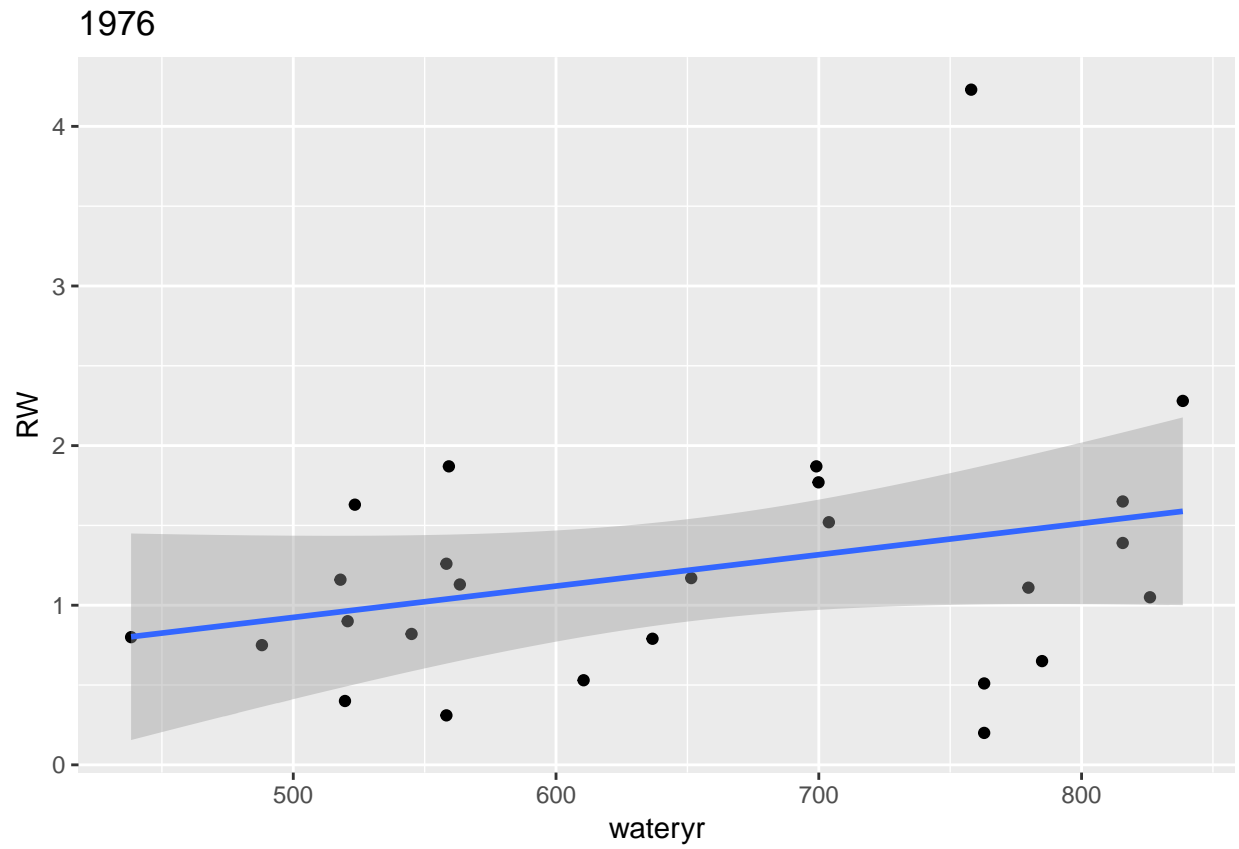
growth_wateryr_plots_es$plots[[1]]
```



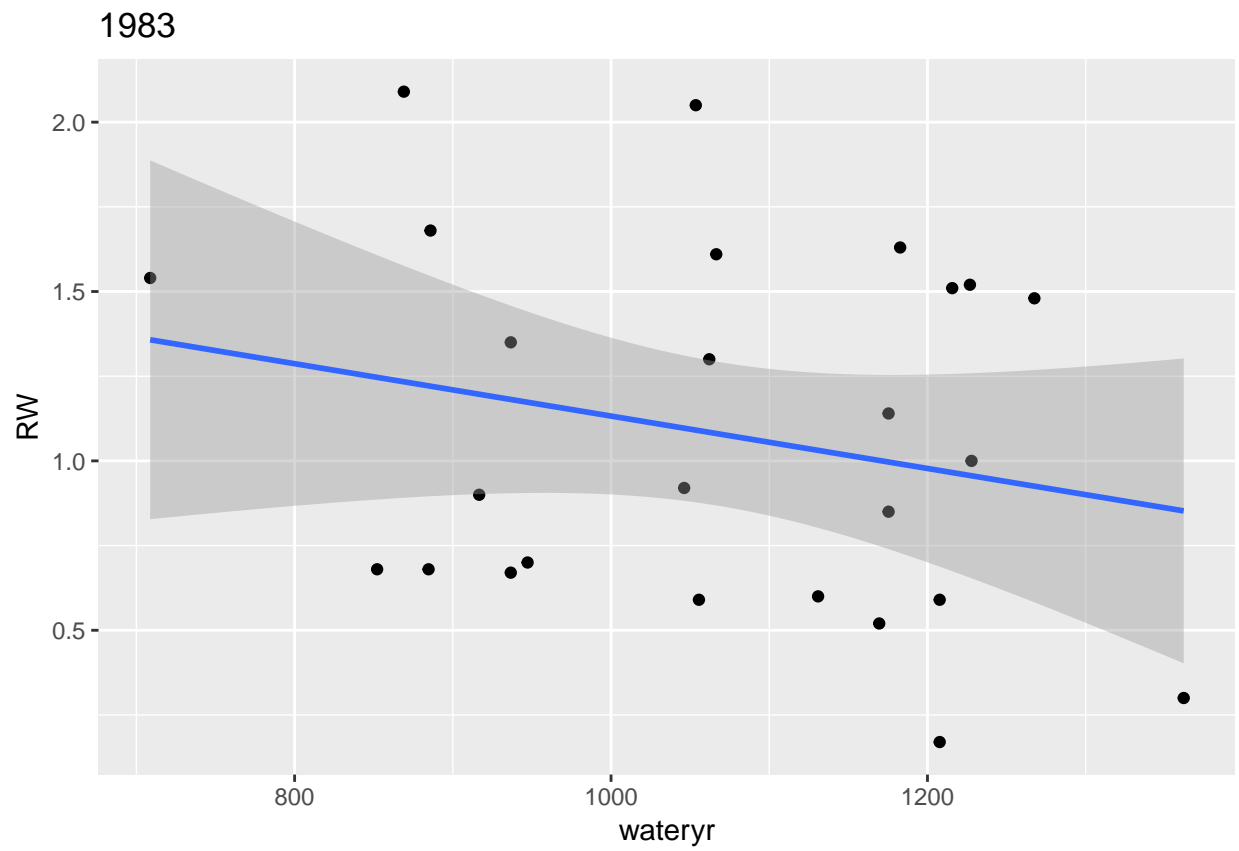
```
growth_wateryr_plots_es$plots[[8]]
```



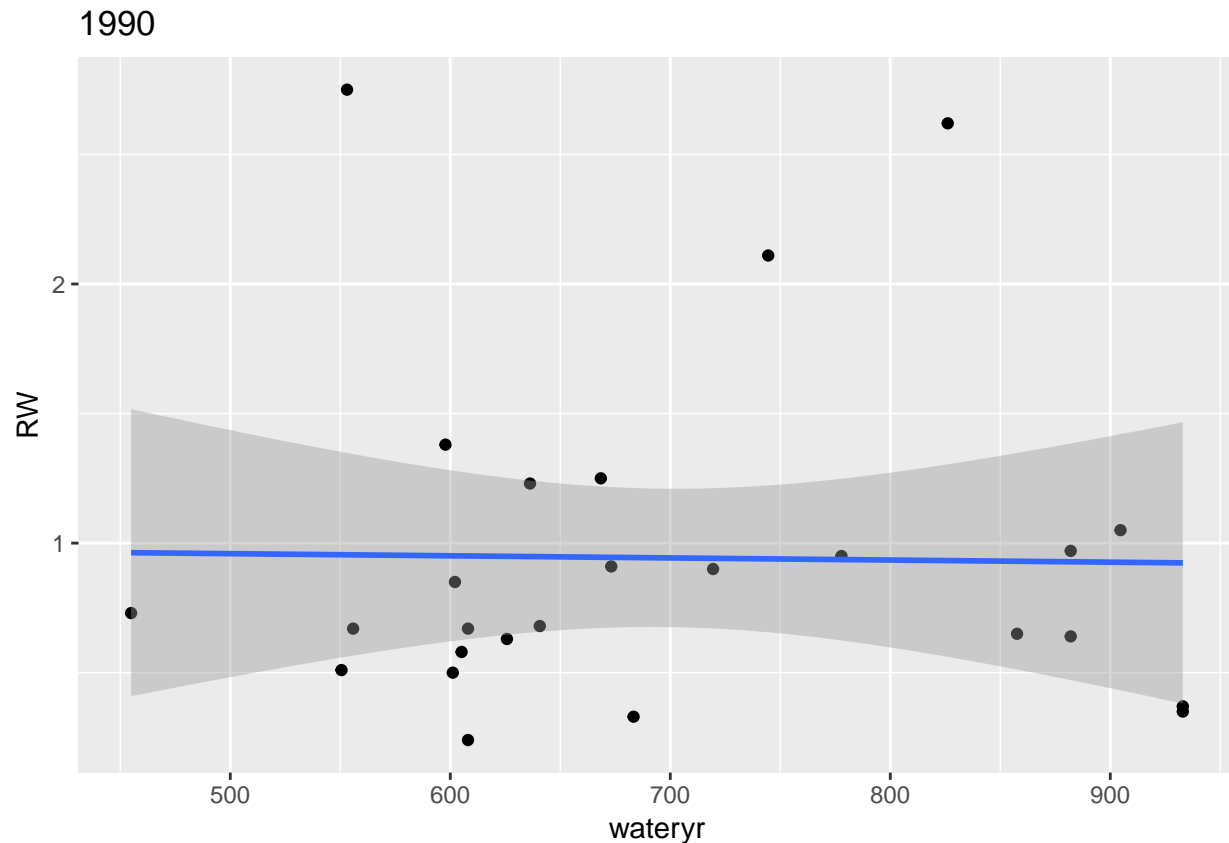
```
growth_wateryr_plots_es$plots[[15]]
```



```
growth_wateryr_plots_es$plots[[22]]
```



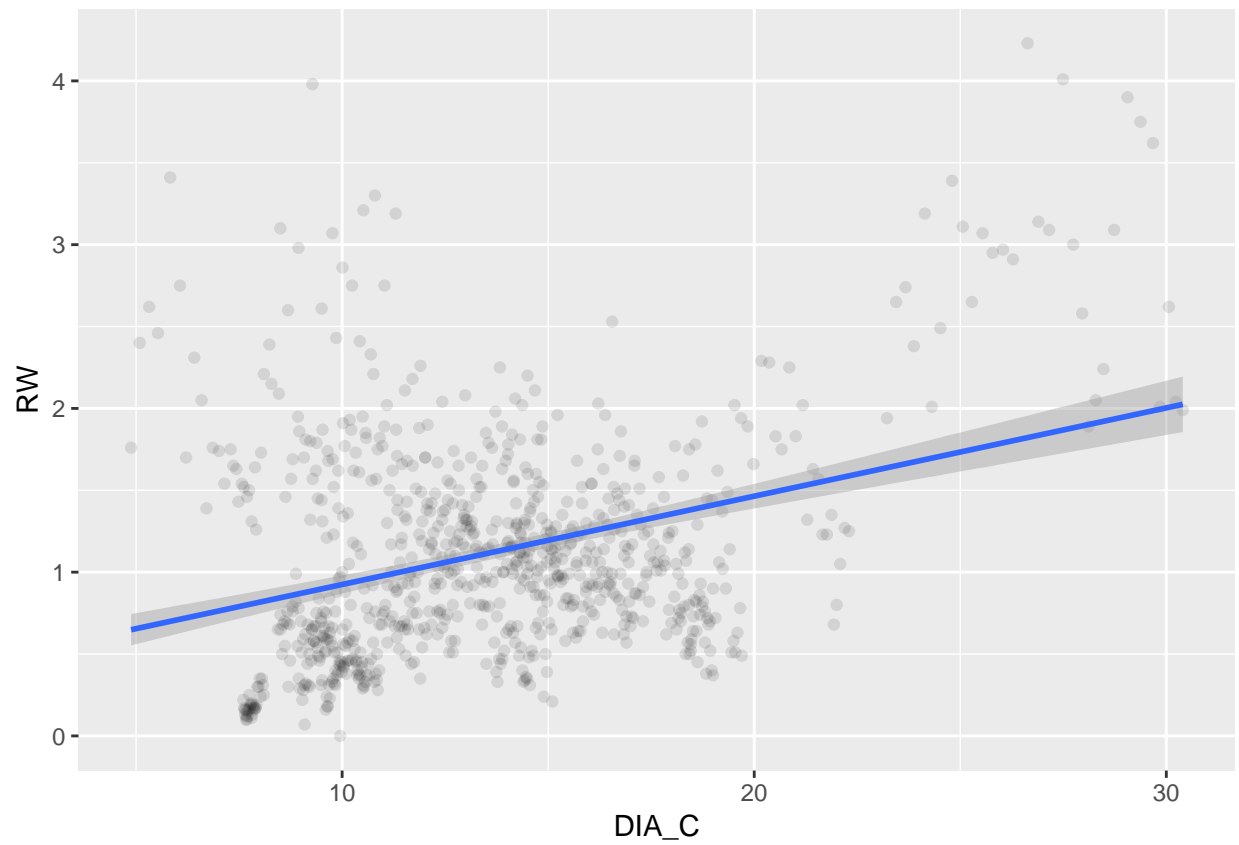
```
growth_wateryr_plots_es$plots[[29]]
```



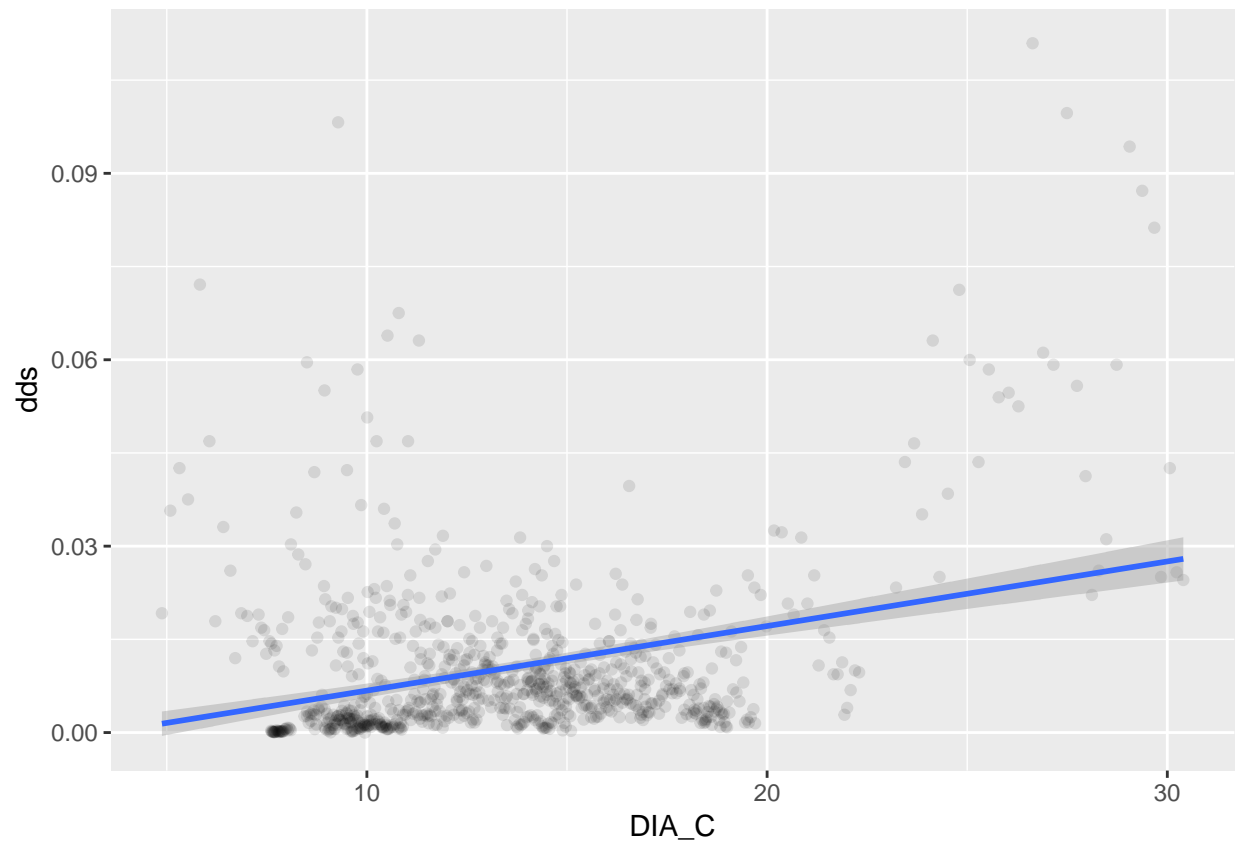
```
#can all years 1962 - 1994
```

A proxy for age (time) is diameter at breast height, which is continuous. As a tree gets older, it grows, puts on biomass, and increases its diameter. While a tree could put on the same amount of biomass each year, the biomass will be stretched over an increasing larger circumference. As such, diameter increment decays as a tree ages, or as diameter at breast height increases. This age trend has been clearly described in tree-ring science. Below is a plot of age (dbh) by my response (incr), which is fit with a linear trend:

```
#detrrend
ggplot(data = glmm_data_es, aes(x = DIA_C, y = RW)) +
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

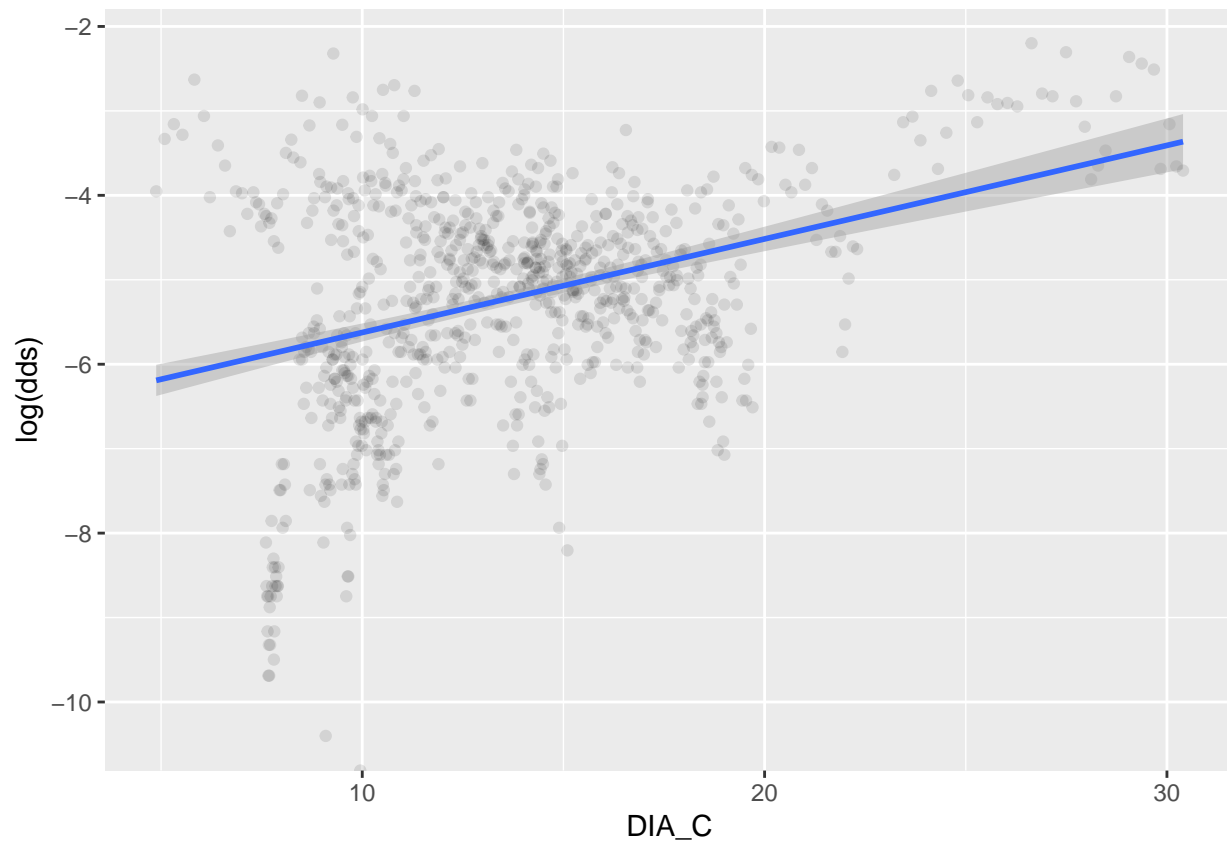



```
ggplot(data = glmm_data_es, aes(x = DIA_C, y = dds)) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```



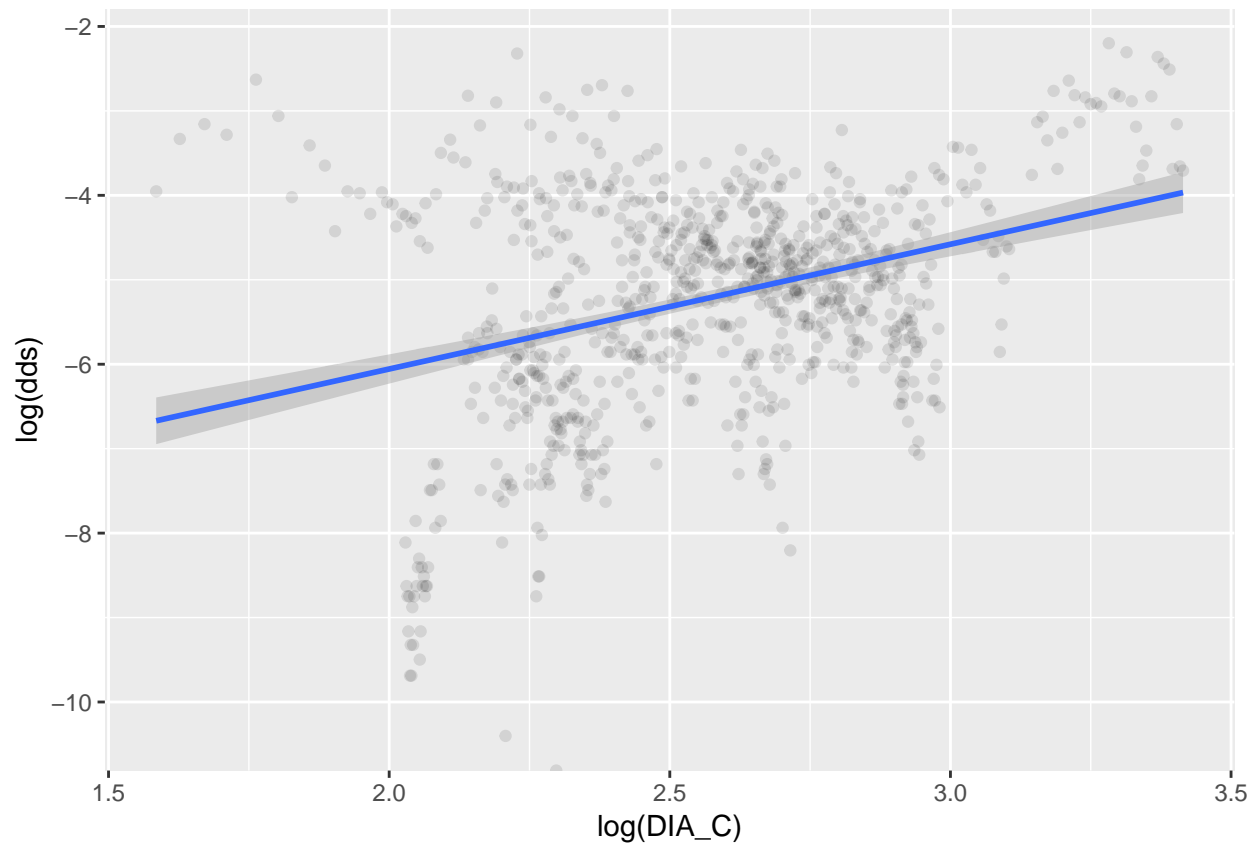
```
ggplot(data = glmm_data_es, aes(x = DIA_C, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```



```
#log to reduce heteroscedasticity  
ggplot(data = glmm_data_es, aes(x = log(DIA_C), y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```



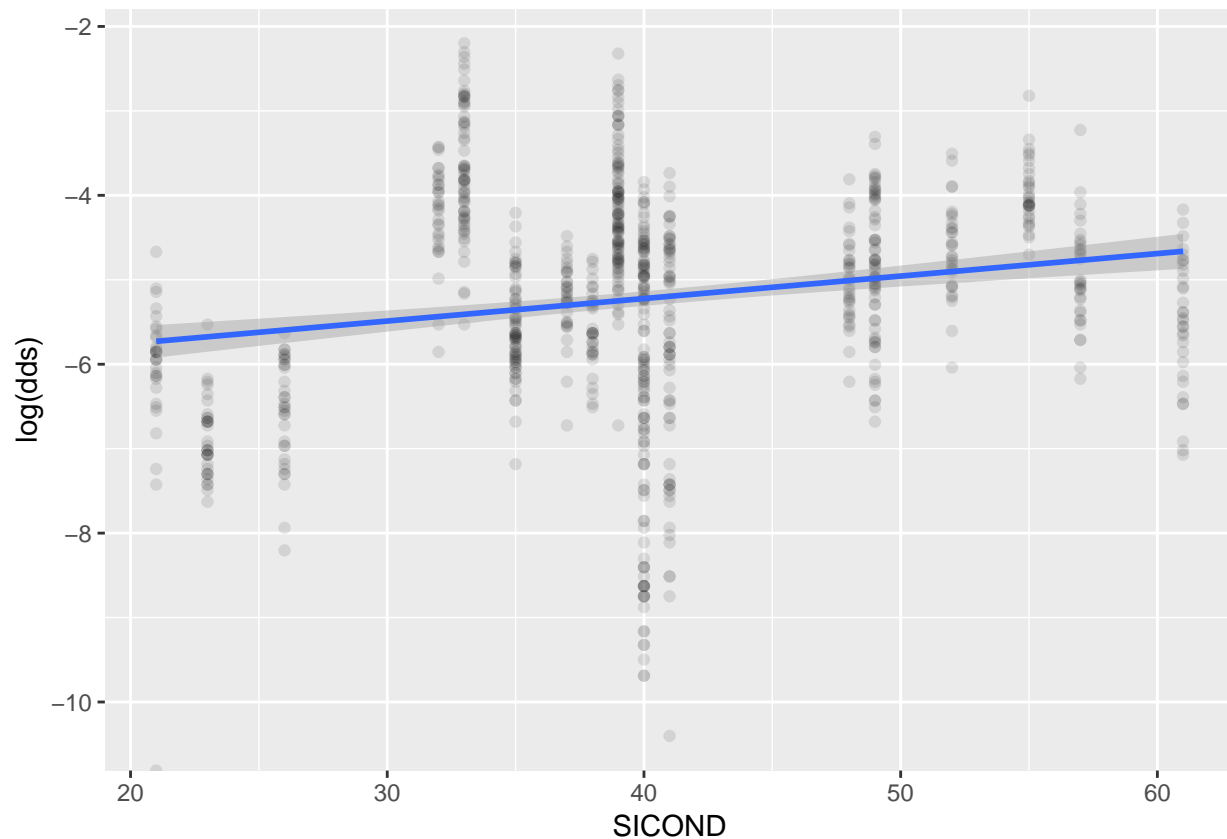
#positive relationship

Log transformation of DIA_C homogenizes variance and allows for linear relationship.

Site index is a measure of site quality. It is the estimated height in feet of a tree on the site at a specific age (FIA Database Description and User Guide for Phase 2).

```
ggplot(data = glmm_data_es, aes(x = SICOND, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

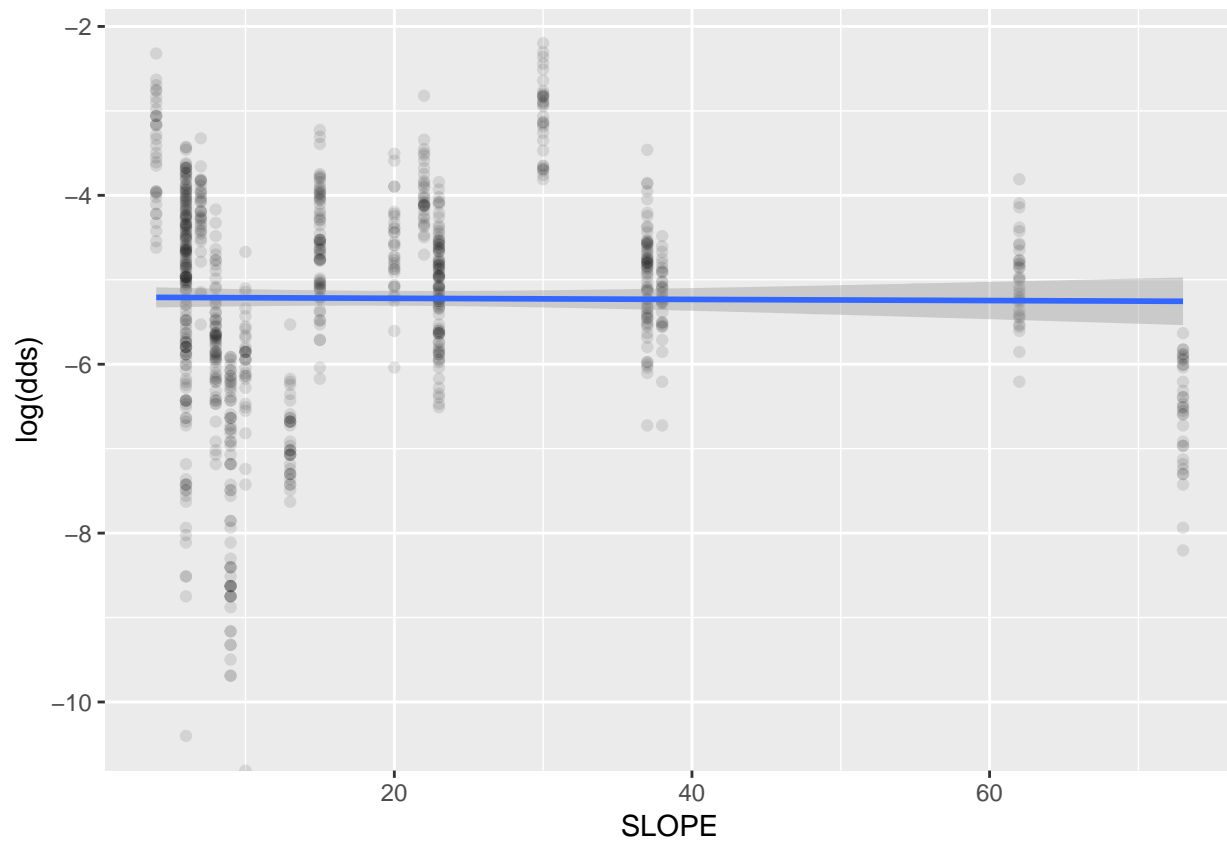


There is a positive relationship between growth and site index, meaning growth increases with increasing site quality. There is also relatively constant variance.

Slope is an estimated average at the subplot level, or condition level (FIA Database Description and User Guide for Phase 2).

```
ggplot(data = glmm_data_es, aes(x = SLOPE, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

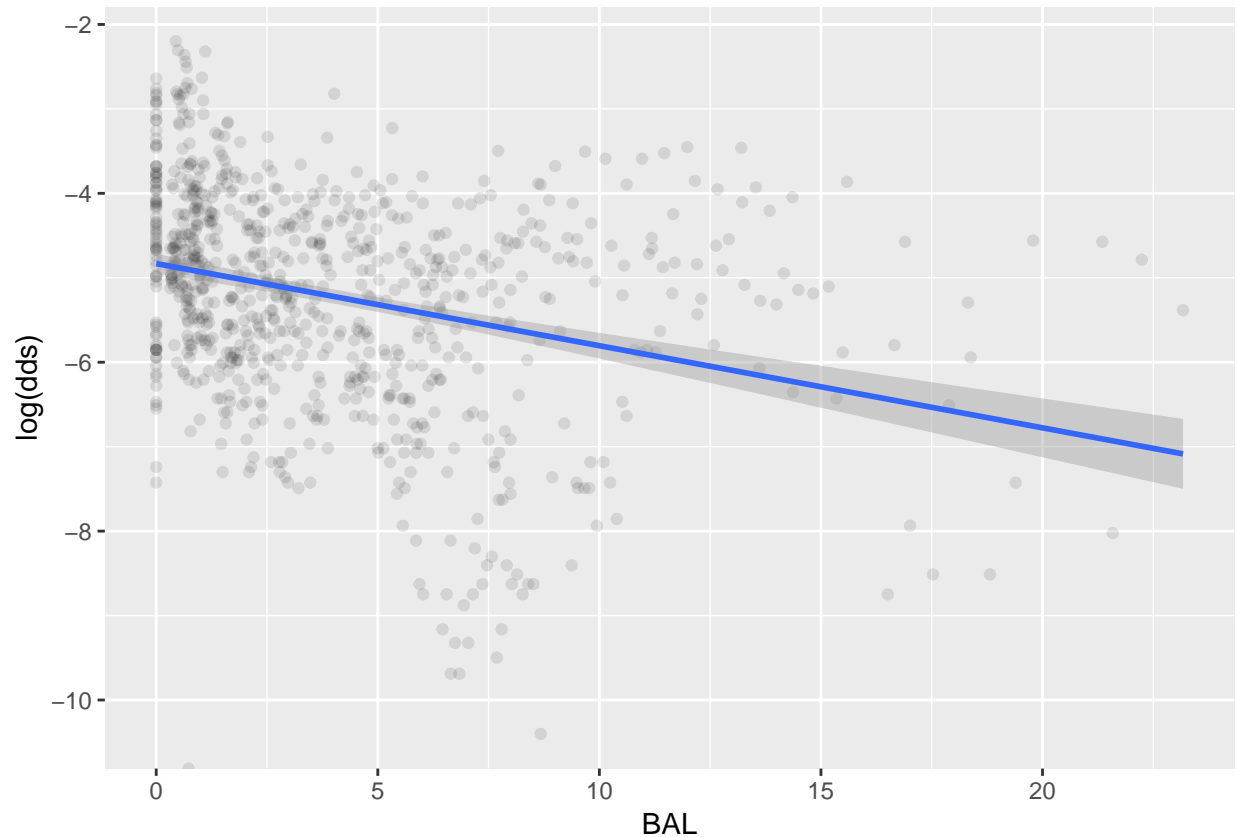


There is no relationship between growth and slope, and there is non-constant variance.

Basal area of trees larger than the subject tree (BAL) is a calculated covariate at the plot level.

```
ggplot(data = glmm_data_es, aes(x = BAL, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```



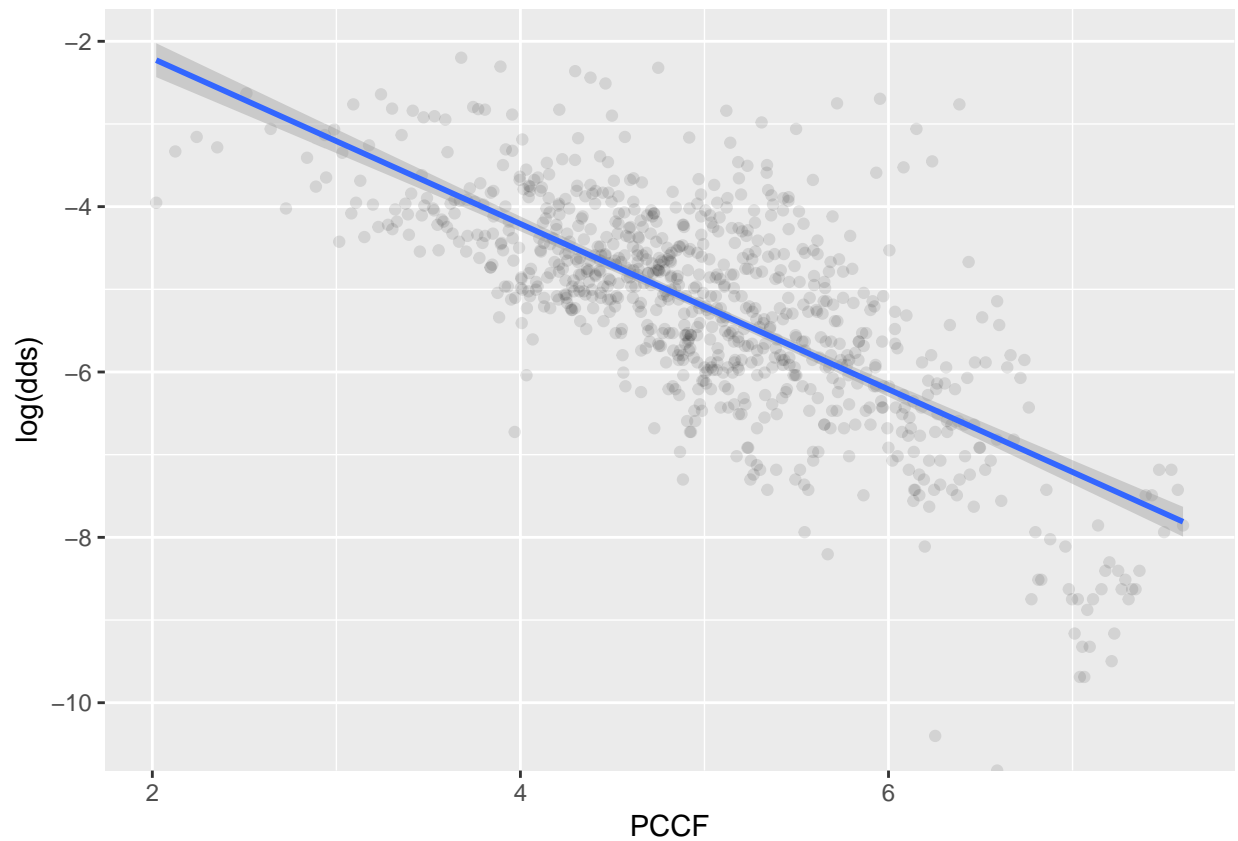
There is a negative relationship between growth and BAL, meaning growth decreases with increasing BAL. There is relatively constant variance.

Crown competition factor is a measure of stand density (Krajicek et al. 1961). Tree values of CCF estimate the percentage of an acre that would be covered by the tree's crown if the tree were open grown (Dixon 2002).

Crown competition factor on the inventory point where the tree is established (PCCF) is a calculated covariate on the plot level.

```
ggplot(data = glmm_data_es, aes(x = PCCF, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

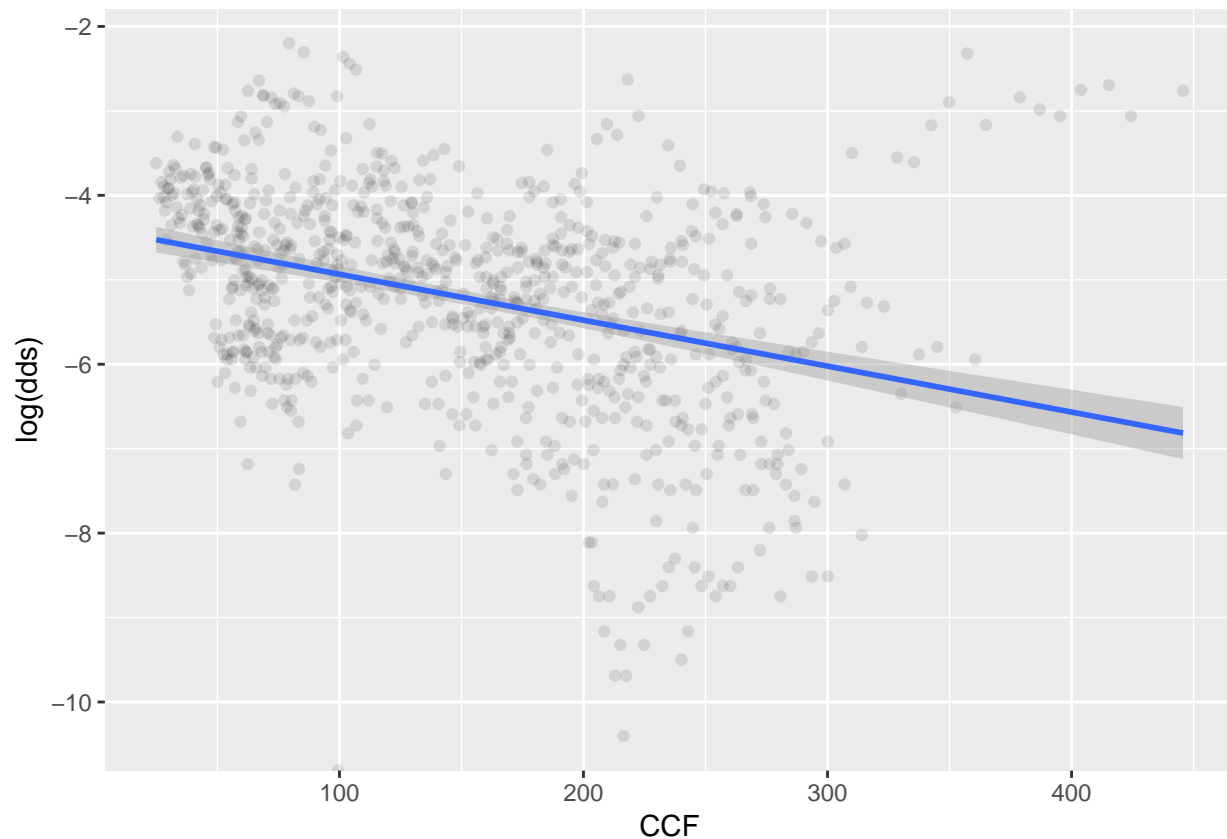


There is a negative relationship between growth and PCCF, and there is relatively constant variance.

Stand crown competition factor (CCF) is a calculated covariate on the plot level. It is a summation of PCCF on the plot.

```
ggplot(data = glmm_data_es, aes(x = CCF, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```

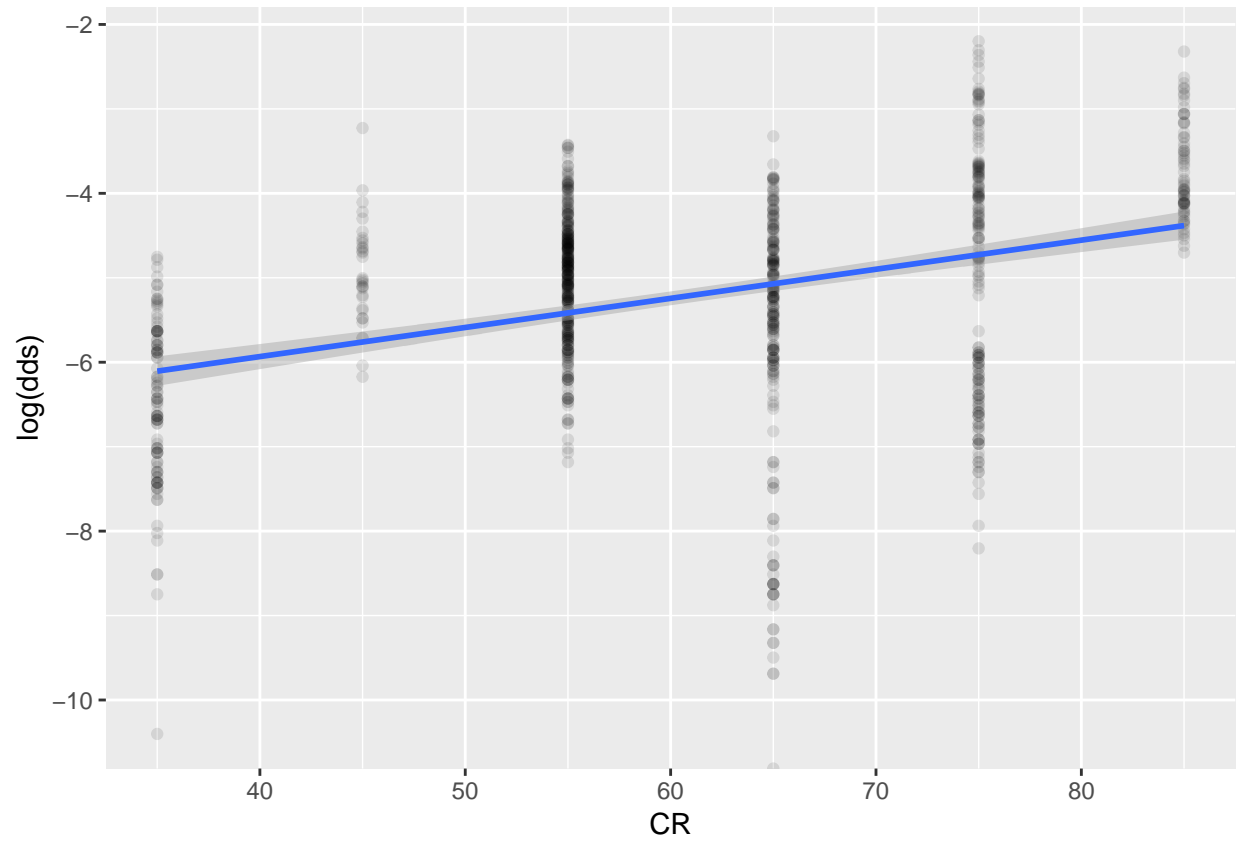



There is a negative relationship between growth and CCF with relatively constant variance.

Crown ratio (CR/CR_weib) is the percentage of the length of the tree with healthy foliage (FIA Database Description and User Guide for Phase 2). CR was extracted from the FIA Database for the measure year and copied backwards. CR_weib was back calculated using the Weibull distribution (Dixon 1985) on a plot level.

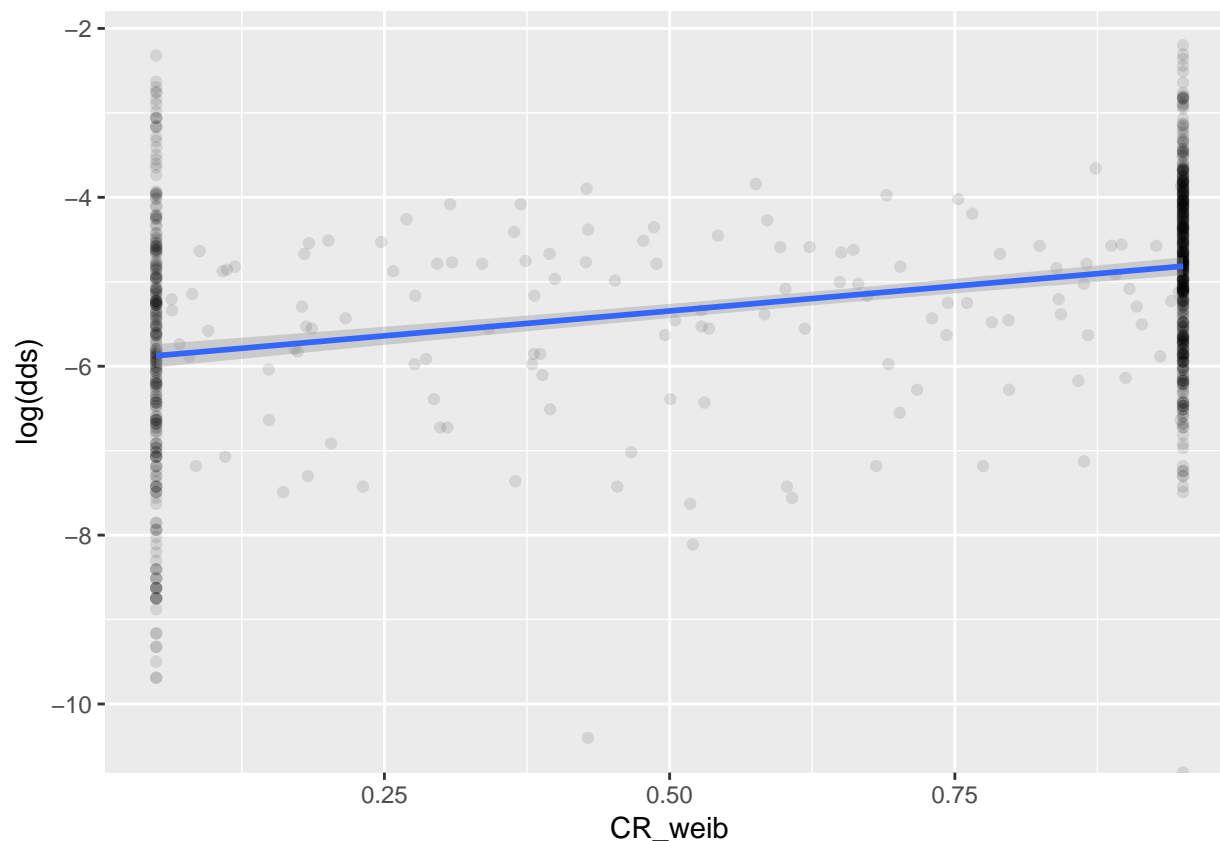
```
ggplot(data = glmm_data_es, aes(x = CR, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```



```
ggplot(data = glmm_data_es, aes(x = CR_weib, y = log(dds))) +  
  geom_point(alpha=0.1) + geom_smooth(method = "lm")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).
```



There is a positive relationship between growth and CR and relatively constant variance. Likewise, there is a positive relationship between growth and CR_weib and constant variance.

Based on the exploration, I will transform the response variable (dds) and DIA_C to linearize the relationship and reduce heteroscedasticity and satisfy assumptions of normality of the response variable. I will also explore outliers with Cleveland dotplots. Normality and homogeneity will be further checked using the residuals of the final models.

Model Building

For now, I will use `lme4` to fit models. To fit a covariance-variance structure other than compound symmetry, I can use `nlme`. I may also explore the model fit options with `glmmTMB`.

The current large diameter growth model for Douglas fir in the Utah variant of FVS is a multiple regression.

```
old_fvs <- lm(log(dds + 0.001) ~ SICOND + I(sin(ASPECT - 0.7854) * SLOPE) +
              I(cos(ASPECT - 0.7854) * SLOPE) + SLOPE + I(log(DIA_C)) +
              I(BAL/100) + CR + I(CR^2) + I(CCF/100), data = glmm_data_es)
summary(old_fvs)$coef
```

	Estimate	Std. Error	t value
## (Intercept)	-7.3350881553	0.4403541876	-16.6572463
## SICOND	0.0035905094	0.0025207979	1.4243543
## I(sin(ASPECT - 0.7854) * SLOPE)	-0.0128637669	0.0015540881	-8.2773732
## I(cos(ASPECT - 0.7854) * SLOPE)	0.0003685403	0.0020289202	0.1816436
## SLOPE	0.0054110151	0.0021301006	2.5402627

```
## I(log(DIA_C))          1.2044786286 0.0961315984 12.5294768
## I(BAL/100)             4.3509669479 0.8096151889 5.3741172
## CR                    -0.0258660157 0.0150352989 -1.7203526
## I(CR^2)                0.0003221156 0.0001279349 2.5178090
## I(CCF/100)            -0.5821628291 0.0420800611 -13.8346479
##                        Pr(>|t|)
## (Intercept)           1.203150e-53
## SCOND                 1.547373e-01
## I(sin(ASPECT - 0.7854) * SLOPE) 5.320069e-16
## I(cos(ASPECT - 0.7854) * SLOPE) 8.559089e-01
## SLOPE                 1.126661e-02
## I(log(DIA_C))         5.457558e-33
## I(BAL/100)            1.012566e-07
## CR                   8.575848e-02
## I(CR^2)              1.200466e-02
## I(CCF/100)          3.760683e-39
```

My growth model using tree ring and forest inventory data will be a linear mixed model.

```
full_model <- lmer(log(dds + 0.001) ~ log(DIA_C) + SCOND + SLOPE + ASPECT +
                  BAL + PCCF + CCF + CR + wateryr + tmin_pNovApr +
                  (1|TRE_CN) + (1|Year), data=glmm_data_es)
summary(full_model)$coef
```

```
##              Estimate Std. Error      df    t value      Pr(>|t|)
## (Intercept) -7.858269e+00 1.0131024796 40.11168 -7.7566380 1.663608e-09
## log(DIA_C)   1.383873e+00 0.3010078696 21.72123 4.5974634 1.443079e-04
## SCOND        1.514718e-03 0.0109885540 18.18221 0.1378451 8.918779e-01
## SLOPE        -7.053680e-03 0.0064394203 19.00273 -1.0953905 2.870352e-01
## ASPECT       1.752544e-04 0.0007933118 18.10489 0.2209149 8.276309e-01
## BAL          3.705710e-02 0.0094873931 611.50460 3.9059306 1.043239e-04
## PCCF         -3.161756e-01 0.0661349443 77.69430 -4.7807651 8.119413e-06
## CCF          -1.975949e-03 0.0009957396 225.95797 -1.9844029 4.842086e-02
## CR           1.716640e-02 0.0085018889 18.65498 2.0191276 5.807964e-02
## wateryr      9.039486e-05 0.0001235281 438.97200 0.7317758 4.646959e-01
## tmin_pNovApr 2.763906e-03 0.0220300351 267.08446 0.1254608 9.002531e-01
```

Collinearity

Using the full model with all the possible covariates, I will test for collinearity between predictors with the car package.

```
##    log(DIA_C)      SCOND      SLOPE      ASPECT      BAL
##    1.399018      1.119802      1.199010      1.104065      1.718970
##    PCCF          CCF          CR          wateryr tmin_pNovApr
##    1.730367      1.814198      1.180211      1.009318      1.055118
```

The Variance Inflation Factors (VIFs) are relatively low, showing no collinearity.

Model Selection

Compare models via Akaike's Information Criterion. Check for homogeneity and normality using residuals.