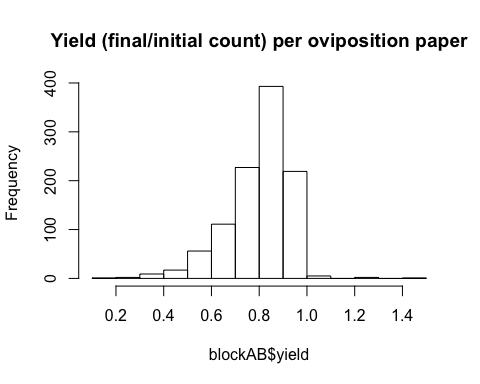
CPP\_analysis

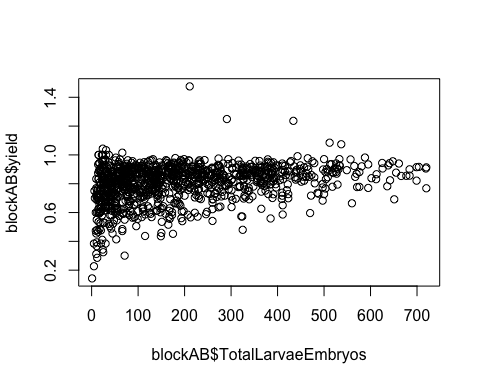
library("drc") #3.0.1  
library("knitr") #1.25  
library("tidyverse") #1.2.1  
library("stringr") #1.4.0  
library("geosphere") #1.5.10  
library("lubridate") #1.7.4

all\_data<-data.frame(read.csv("CPP\_experiment\_data\_collated.csv"))  
#Filter out empty/discarded oviposition papers   
blockAB<-dplyr::filter(all\_data, TotalLarvaeEmbryos>0)

#Calculate yield/fraction of initial count  
blockAB$yield <- blockAB$TotalLarvaeEmbryos/blockAB$InitialEggs   
hist(blockAB$yield, main="Yield (final/initial count) per oviposition paper")



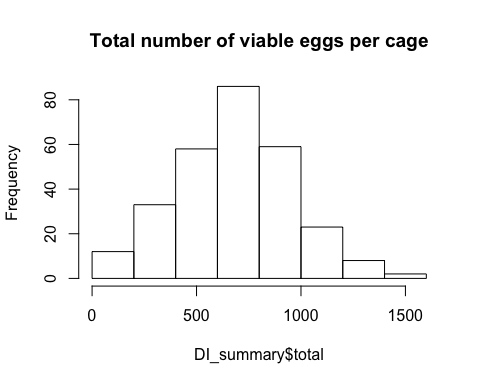
plot(blockAB$yield ~ blockAB$TotalLarvaeEmbryos)



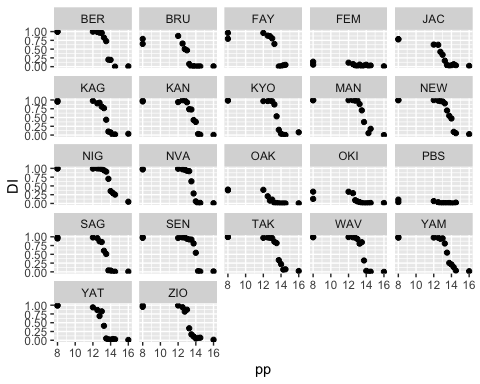
#removing papers with <60% or >100% initial count  
blockAB <- filter(blockAB, yield<=1) %>% filter(yield>=0.6)

DI\_summary <- blockAB %>%   
 group\_by(CageID) %>%   
 summarise(pop=first(Population), pp=mean(Photoperiod), lat=mean(Lat), block=first(Block), country=first(Country), h1=sum(H1Count), h2=sum(H2Count), emb=sum(EmbryoCount)) %>%   
 filter(h1+h2+emb>100)  
DI\_summary$total<-DI\_summary$h1+DI\_summary$h2+DI\_summary$emb  
DI\_summary$DI<-DI\_summary$emb/DI\_summary$total

hist(DI\_summary$total, main="Total number of viable eggs per cage")

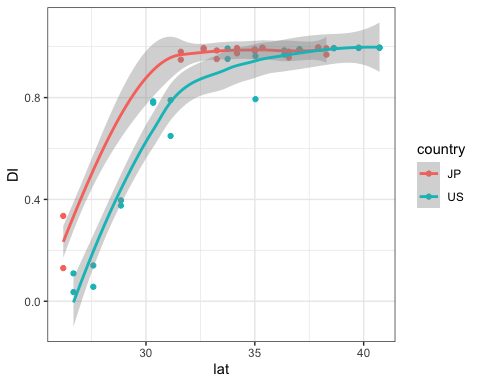


#plot with DI at 8 h shown as TWO DIFFERENT points (corresponding to two cabinets)  
ggplot(DI\_summary, aes(x=pp, y=DI, group=pop)) + geom\_point() + facet\_wrap(~pop)



DI\_short<-filter(DI\_summary, pp==8)  
ggplot(DI\_short, aes(x=lat, y=DI, color=country)) + geom\_point() + theme\_bw() +stat\_smooth()

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



# Calculating CPP values

5-parameter generalized log-logistic model. Explanation on page 2: <http://ftp.uni-bayreuth.de/math/statlib/R/CRAN/doc/vignettes/drc/drc.pdf> d and c are upper and lower limit, respectively e is ED50 b is relative slope aroung e f represents asymmetry in the model; f=1 if function is symmetrical

#From Kevin Emerson  
  
# the following fits the models of the given function using data from the  
# data.frame d separately for the various treatments using a 5 parameter  
# logistic model  
d.model.fit <- drm(DI ~ pp, data = DI\_summary, curveid = pop, fct = LL.5())  
#plot(d.model.fit, log = "")  
  
# output the parameters of the model fits:  
summary(d.model.fit)

## Warning in sqrt(diag(varMat)): NaNs produced

##   
## Model fitted: Generalized log-logistic (ED50 as parameter) (5 parms)  
##   
## Parameter estimates:  
##   
## Estimate Std. Error t-value p-value   
## b:BER 8.5200e+01 3.2181e+01 2.6475 0.0088663 \*\*   
## b:BRU 5.2790e+01 1.0817e+01 4.8802 2.415e-06 \*\*\*  
## b:KAG 6.6054e+01 1.7062e+01 3.8715 0.0001537 \*\*\*  
## b:KAN 3.8259e+01 7.5934e+00 5.0384 1.185e-06 \*\*\*  
## b:KYO 1.0496e+02 4.0213e+01 2.6102 0.0098517 \*\*   
## b:NVA 9.1447e+01 8.5834e+01 1.0654 0.2881999   
## b:OAK 1.0185e+02 1.0456e+02 0.9740 0.3314148   
## b:SAG 5.1860e+01 9.1496e+00 5.6681 6.023e-08 \*\*\*  
## b:SEN 9.7383e+01 2.2790e+01 4.2731 3.194e-05 \*\*\*  
## b:WAV 9.8219e+01 3.2422e+01 3.0294 0.0028305 \*\*   
## b:PBS 1.8725e+01 3.7370e+01 0.5011 0.6169695   
## b:FEM 2.4644e+02 2.1399e+03 0.1152 0.9084492   
## b:JAC 3.1863e+01 7.5510e+00 4.2197 3.962e-05 \*\*\*  
## b:MAN 6.6063e+01 3.5770e+01 1.8469 0.0664932 .   
## b:NEW 3.7218e+01 5.6668e+00 6.5678 5.896e-10 \*\*\*  
## b:NIG 5.9342e+01 2.0370e+01 2.9132 0.0040560 \*\*   
## b:OKI 3.8877e+02 4.7716e+02 0.8148 0.4163419   
## b:TAK 2.1390e+02 NA NA NA   
## b:YAM 1.8569e+02 NA NA NA   
## b:YAT 6.9317e+01 2.2448e+01 3.0878 0.0023531 \*\*   
## b:ZIO 3.4000e+02 3.4339e+02 0.9901 0.3235134   
## b:FAY 8.8067e+01 3.5773e+01 2.4618 0.0148154 \*   
## c:BER 1.8110e-02 4.3769e-02 0.4138 0.6795602   
## c:BRU 1.7145e-03 2.1600e-02 0.0794 0.9368266   
## c:KAG 2.7917e-02 2.7381e-02 1.0196 0.3093737   
## c:KAN -1.8262e-02 4.2849e-02 -0.4262 0.6705078   
## c:KYO 3.3159e-02 2.7856e-02 1.1904 0.2355426   
## c:NVA 3.8830e-03 4.3294e-02 0.0897 0.9286400   
## c:OAK -1.2484e-03 2.7284e-02 -0.0458 0.9635582   
## c:SAG -4.7856e-03 2.7649e-02 -0.1731 0.8627871   
## c:SEN -1.6296e-03 3.4372e-02 -0.0474 0.9622404   
## c:WAV 3.7112e-04 2.8477e-02 0.0130 0.9896172   
## c:PBS -1.3042e-04 NA NA NA   
## c:FEM 2.9318e-02 1.7221e-02 1.7024 0.0904892 .   
## c:JAC 3.2054e-02 2.3522e-02 1.3627 0.1747608   
## c:MAN 9.1790e-02 3.0068e-02 3.0527 0.0026304 \*\*   
## c:NEW 1.0853e-02 4.3204e-02 0.2512 0.8019495   
## c:NIG 1.6660e-01 4.7217e-02 3.5283 0.0005372 \*\*\*  
## c:OKI 2.0233e-02 1.8277e-02 1.1070 0.2698590   
## c:TAK 3.8324e-02 3.7576e-02 1.0199 0.3092078   
## c:YAM -9.2306e-04 3.9942e-02 -0.0231 0.9815896   
## c:YAT 1.7212e-02 2.3058e-02 0.7465 0.4564115   
## c:ZIO 5.0581e-02 2.8415e-02 1.7801 0.0768403 .   
## c:FAY 2.0792e-02 2.7195e-02 0.7645 0.4456020   
## d:BER 9.8536e-01 2.3992e-02 41.0707 < 2.2e-16 \*\*\*  
## d:BRU 7.5754e-01 3.0722e-02 24.6579 < 2.2e-16 \*\*\*  
## d:KAG 9.4474e-01 2.5584e-02 36.9266 < 2.2e-16 \*\*\*  
## d:KAN 9.8005e-01 2.5532e-02 38.3855 < 2.2e-16 \*\*\*  
## d:KYO 9.7957e-01 2.2315e-02 43.8978 < 2.2e-16 \*\*\*  
## d:NVA 9.7410e-01 2.6072e-02 37.3627 < 2.2e-16 \*\*\*  
## d:OAK 3.9080e-01 3.2787e-02 11.9195 < 2.2e-16 \*\*\*  
## d:SAG 9.6258e-01 2.6957e-02 35.7084 < 2.2e-16 \*\*\*  
## d:SEN 9.5988e-01 1.9261e-02 49.8345 < 2.2e-16 \*\*\*  
## d:WAV 9.6582e-01 2.2242e-02 43.4226 < 2.2e-16 \*\*\*  
## d:PBS 7.5313e-02 3.4741e-02 2.1679 0.0315524 \*   
## d:FEM 1.0345e-01 2.9845e-02 3.4663 0.0006673 \*\*\*  
## d:JAC 7.6098e-01 3.5713e-02 21.3082 < 2.2e-16 \*\*\*  
## d:MAN 9.9575e-01 2.4627e-02 40.4326 < 2.2e-16 \*\*\*  
## d:NEW 1.0009e+00 2.4922e-02 40.1628 < 2.2e-16 \*\*\*  
## d:NIG 1.0007e+00 2.3213e-02 43.1083 < 2.2e-16 \*\*\*  
## d:OKI 2.7342e-01 2.5851e-02 10.5767 < 2.2e-16 \*\*\*  
## d:TAK 9.6412e-01 1.4024e-02 68.7482 < 2.2e-16 \*\*\*  
## d:YAM 9.7049e-01 2.0617e-02 47.0713 < 2.2e-16 \*\*\*  
## d:YAT 9.2628e-01 3.2201e-02 28.7654 < 2.2e-16 \*\*\*  
## d:ZIO 9.3999e-01 2.3065e-02 40.7547 < 2.2e-16 \*\*\*  
## d:FAY 8.9327e-01 2.4571e-02 36.3544 < 2.2e-16 \*\*\*  
## e:BER 1.3583e+01 2.1044e-01 64.5467 < 2.2e-16 \*\*\*  
## e:BRU 1.3508e+01 4.1065e-01 32.8946 < 2.2e-16 \*\*\*  
## e:KAG 1.3739e+01 3.5443e-01 38.7644 < 2.2e-16 \*\*\*  
## e:KAN 1.4370e+01 8.1369e-01 17.6604 < 2.2e-16 \*\*\*  
## e:KYO 1.3545e+01 1.9818e-01 68.3444 < 2.2e-16 \*\*\*  
## e:NVA 1.3605e+01 5.9121e-01 23.0125 < 2.2e-16 \*\*\*  
## e:OAK 1.2252e+01 2.3988e-01 51.0752 < 2.2e-16 \*\*\*  
## e:SAG 1.3842e+01 3.2668e-01 42.3713 < 2.2e-16 \*\*\*  
## e:SEN 1.4259e+01 2.0668e-01 68.9918 < 2.2e-16 \*\*\*  
## e:WAV 1.3835e+01 2.2243e-01 62.1972 < 2.2e-16 \*\*\*  
## e:PBS 1.2948e+01 4.6632e+00 2.7765 0.0061068 \*\*   
## e:FEM 1.2636e+01 1.4547e+00 8.6864 2.899e-15 \*\*\*  
## e:JAC 1.3716e+01 8.0583e-01 17.0205 < 2.2e-16 \*\*\*  
## e:MAN 1.4025e+01 1.9329e+00 7.2558 1.333e-11 \*\*\*  
## e:NEW 1.4687e+01 7.3310e-01 20.0339 < 2.2e-16 \*\*\*  
## e:NIG 1.4197e+01 NA NA NA   
## e:OKI 1.2741e+01 NA NA NA   
## e:TAK 1.3486e+01 NA NA NA   
## e:YAM 1.3202e+01 NA NA NA   
## e:YAT 1.3538e+01 2.7596e-01 49.0565 < 2.2e-16 \*\*\*  
## e:ZIO 1.3014e+01 7.0766e-02 183.8946 < 2.2e-16 \*\*\*  
## e:FAY 1.3547e+01 3.0099e-01 45.0092 < 2.2e-16 \*\*\*  
## f:BER 9.2641e-01 8.8530e-01 1.0464 0.2968363   
## f:BRU 5.7815e+00 7.6157e+00 0.7592 0.4488031   
## f:KAG 3.0402e+00 3.9448e+00 0.7707 0.4419627   
## f:KAN 3.9370e+00 6.7975e+00 0.5792 0.5632271   
## f:KYO 1.1956e+00 1.2554e+00 0.9524 0.3422622   
## f:NVA 1.0129e+00 2.7457e+00 0.3689 0.7126622   
## f:OAK 3.0568e-01 4.0268e-01 0.7591 0.4488320   
## f:SAG 3.6185e+00 3.4930e+00 1.0359 0.3017037   
## f:SEN 4.0810e+00 4.3820e+00 0.9313 0.3530037   
## f:WAV 2.3921e+00 2.7275e+00 0.8770 0.3817103   
## f:PBS 2.0375e+00 4.6436e+00 0.4388 0.6613828   
## f:FEM 7.3344e+00 2.0275e+02 0.0362 0.9711849   
## f:JAC 5.8284e+00 8.6409e+00 0.6745 0.5009001   
## f:MAN 4.9730e+00 3.7221e+01 0.1336 0.8938710   
## f:NEW 6.5840e+00 1.0316e+01 0.6382 0.5241845   
## f:NIG 3.3452e+00 NA NA NA   
## f:OKI 1.4907e+00 NA NA NA   
## f:TAK 2.4460e-01 NA NA NA   
## f:YAM 1.5278e-01 NA NA NA   
## f:YAT 4.1355e+00 4.1496e+00 0.9966 0.3203711   
## f:ZIO 1.7205e-01 2.1578e-01 0.7973 0.4263594   
## f:FAY 2.6505e+00 3.4205e+00 0.7749 0.4394729   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error:  
##   
## 0.05163599 (171 degrees of freedom)

# estimate the 50% intercept  
ED(d.model.fit, 50)

## Warning in sqrt(dEDval %\*% varCov %\*% dEDval): NaNs produced

## Warning in sqrt(dEDval %\*% varCov %\*% dEDval): NaNs produced

##   
## Estimated effective doses  
##   
## Estimate Std. Error  
## e:BER:50 13.600154 0.032426  
## e:BRU:50 12.991086 0.045615  
## e:FAY:50 13.362601 0.049503  
## e:FEM:50 12.517899 0.204107  
## e:JAC:50 12.853244 0.069543  
## e:KAG:50 13.458850 0.032953  
## e:KAN:50 13.764317 0.049295  
## e:KYO:50 13.513563 0.027389  
## e:MAN:50 13.627346 0.040470  
## e:NEW:50 13.844557 0.049230  
## e:NIG:50 13.849662 0.049069  
## e:NVA:50 13.602618 0.043421  
## e:OAK:50 12.514155 0.110016  
## e:OKI:50 12.723369 0.067729  
## e:PBS:50 12.337784 0.978213  
## e:SAG:50 13.432874 0.037531  
## e:SEN:50 14.014586 0.024896  
## e:TAK:50 13.662345 NA  
## e:WAV:50 13.682082 0.024563  
## e:YAM:50 13.527540 0.042665  
## e:YAT:50 13.209568 0.034621  
## e:ZIO:50 13.167965 0.034344

cpp\_values<-data.frame(ED(d.model.fit, 50))

## Warning in sqrt(dEDval %\*% varCov %\*% dEDval): NaNs produced  
  
## Warning in sqrt(dEDval %\*% varCov %\*% dEDval): NaNs produced

##   
## Estimated effective doses  
##   
## Estimate Std. Error  
## e:BER:50 13.600154 0.032426  
## e:BRU:50 12.991086 0.045615  
## e:FAY:50 13.362601 0.049503  
## e:FEM:50 12.517899 0.204107  
## e:JAC:50 12.853244 0.069543  
## e:KAG:50 13.458850 0.032953  
## e:KAN:50 13.764317 0.049295  
## e:KYO:50 13.513563 0.027389  
## e:MAN:50 13.627346 0.040470  
## e:NEW:50 13.844557 0.049230  
## e:NIG:50 13.849662 0.049069  
## e:NVA:50 13.602618 0.043421  
## e:OAK:50 12.514155 0.110016  
## e:OKI:50 12.723369 0.067729  
## e:PBS:50 12.337784 0.978213  
## e:SAG:50 13.432874 0.037531  
## e:SEN:50 14.014586 0.024896  
## e:TAK:50 13.662345 NA  
## e:WAV:50 13.682082 0.024563  
## e:YAM:50 13.527540 0.042665  
## e:YAT:50 13.209568 0.034621  
## e:ZIO:50 13.167965 0.034344

# Recalculating CPP values for 2008 photoperiod response curves

DI\_summary\_2008<-data.frame(read.csv("CPP\_2008.csv"))  
d.model.fit.2008 <- drm(DI ~ photoperiod, data = DI\_summary\_2008, curveid = pop, fct = LL.5())  
   
#plot(d.model.fit.2008, log = "")  
# output the parameters of the model fits:  
summary(d.model.fit)

## Warning in sqrt(diag(varMat)): NaNs produced

##   
## Model fitted: Generalized log-logistic (ED50 as parameter) (5 parms)  
##   
## Parameter estimates:  
##   
## Estimate Std. Error t-value p-value   
## b:BER 8.5200e+01 3.2181e+01 2.6475 0.0088663 \*\*   
## b:BRU 5.2790e+01 1.0817e+01 4.8802 2.415e-06 \*\*\*  
## b:KAG 6.6054e+01 1.7062e+01 3.8715 0.0001537 \*\*\*  
## b:KAN 3.8259e+01 7.5934e+00 5.0384 1.185e-06 \*\*\*  
## b:KYO 1.0496e+02 4.0213e+01 2.6102 0.0098517 \*\*   
## b:NVA 9.1447e+01 8.5834e+01 1.0654 0.2881999   
## b:OAK 1.0185e+02 1.0456e+02 0.9740 0.3314148   
## b:SAG 5.1860e+01 9.1496e+00 5.6681 6.023e-08 \*\*\*  
## b:SEN 9.7383e+01 2.2790e+01 4.2731 3.194e-05 \*\*\*  
## b:WAV 9.8219e+01 3.2422e+01 3.0294 0.0028305 \*\*   
## b:PBS 1.8725e+01 3.7370e+01 0.5011 0.6169695   
## b:FEM 2.4644e+02 2.1399e+03 0.1152 0.9084492   
## b:JAC 3.1863e+01 7.5510e+00 4.2197 3.962e-05 \*\*\*  
## b:MAN 6.6063e+01 3.5770e+01 1.8469 0.0664932 .   
## b:NEW 3.7218e+01 5.6668e+00 6.5678 5.896e-10 \*\*\*  
## b:NIG 5.9342e+01 2.0370e+01 2.9132 0.0040560 \*\*   
## b:OKI 3.8877e+02 4.7716e+02 0.8148 0.4163419   
## b:TAK 2.1390e+02 NA NA NA   
## b:YAM 1.8569e+02 NA NA NA   
## b:YAT 6.9317e+01 2.2448e+01 3.0878 0.0023531 \*\*   
## b:ZIO 3.4000e+02 3.4339e+02 0.9901 0.3235134   
## b:FAY 8.8067e+01 3.5773e+01 2.4618 0.0148154 \*   
## c:BER 1.8110e-02 4.3769e-02 0.4138 0.6795602   
## c:BRU 1.7145e-03 2.1600e-02 0.0794 0.9368266   
## c:KAG 2.7917e-02 2.7381e-02 1.0196 0.3093737   
## c:KAN -1.8262e-02 4.2849e-02 -0.4262 0.6705078   
## c:KYO 3.3159e-02 2.7856e-02 1.1904 0.2355426   
## c:NVA 3.8830e-03 4.3294e-02 0.0897 0.9286400   
## c:OAK -1.2484e-03 2.7284e-02 -0.0458 0.9635582   
## c:SAG -4.7856e-03 2.7649e-02 -0.1731 0.8627871   
## c:SEN -1.6296e-03 3.4372e-02 -0.0474 0.9622404   
## c:WAV 3.7112e-04 2.8477e-02 0.0130 0.9896172   
## c:PBS -1.3042e-04 NA NA NA   
## c:FEM 2.9318e-02 1.7221e-02 1.7024 0.0904892 .   
## c:JAC 3.2054e-02 2.3522e-02 1.3627 0.1747608   
## c:MAN 9.1790e-02 3.0068e-02 3.0527 0.0026304 \*\*   
## c:NEW 1.0853e-02 4.3204e-02 0.2512 0.8019495   
## c:NIG 1.6660e-01 4.7217e-02 3.5283 0.0005372 \*\*\*  
## c:OKI 2.0233e-02 1.8277e-02 1.1070 0.2698590   
## c:TAK 3.8324e-02 3.7576e-02 1.0199 0.3092078   
## c:YAM -9.2306e-04 3.9942e-02 -0.0231 0.9815896   
## c:YAT 1.7212e-02 2.3058e-02 0.7465 0.4564115   
## c:ZIO 5.0581e-02 2.8415e-02 1.7801 0.0768403 .   
## c:FAY 2.0792e-02 2.7195e-02 0.7645 0.4456020   
## d:BER 9.8536e-01 2.3992e-02 41.0707 < 2.2e-16 \*\*\*  
## d:BRU 7.5754e-01 3.0722e-02 24.6579 < 2.2e-16 \*\*\*  
## d:KAG 9.4474e-01 2.5584e-02 36.9266 < 2.2e-16 \*\*\*  
## d:KAN 9.8005e-01 2.5532e-02 38.3855 < 2.2e-16 \*\*\*  
## d:KYO 9.7957e-01 2.2315e-02 43.8978 < 2.2e-16 \*\*\*  
## d:NVA 9.7410e-01 2.6072e-02 37.3627 < 2.2e-16 \*\*\*  
## d:OAK 3.9080e-01 3.2787e-02 11.9195 < 2.2e-16 \*\*\*  
## d:SAG 9.6258e-01 2.6957e-02 35.7084 < 2.2e-16 \*\*\*  
## d:SEN 9.5988e-01 1.9261e-02 49.8345 < 2.2e-16 \*\*\*  
## d:WAV 9.6582e-01 2.2242e-02 43.4226 < 2.2e-16 \*\*\*  
## d:PBS 7.5313e-02 3.4741e-02 2.1679 0.0315524 \*   
## d:FEM 1.0345e-01 2.9845e-02 3.4663 0.0006673 \*\*\*  
## d:JAC 7.6098e-01 3.5713e-02 21.3082 < 2.2e-16 \*\*\*  
## d:MAN 9.9575e-01 2.4627e-02 40.4326 < 2.2e-16 \*\*\*  
## d:NEW 1.0009e+00 2.4922e-02 40.1628 < 2.2e-16 \*\*\*  
## d:NIG 1.0007e+00 2.3213e-02 43.1083 < 2.2e-16 \*\*\*  
## d:OKI 2.7342e-01 2.5851e-02 10.5767 < 2.2e-16 \*\*\*  
## d:TAK 9.6412e-01 1.4024e-02 68.7482 < 2.2e-16 \*\*\*  
## d:YAM 9.7049e-01 2.0617e-02 47.0713 < 2.2e-16 \*\*\*  
## d:YAT 9.2628e-01 3.2201e-02 28.7654 < 2.2e-16 \*\*\*  
## d:ZIO 9.3999e-01 2.3065e-02 40.7547 < 2.2e-16 \*\*\*  
## d:FAY 8.9327e-01 2.4571e-02 36.3544 < 2.2e-16 \*\*\*  
## e:BER 1.3583e+01 2.1044e-01 64.5467 < 2.2e-16 \*\*\*  
## e:BRU 1.3508e+01 4.1065e-01 32.8946 < 2.2e-16 \*\*\*  
## e:KAG 1.3739e+01 3.5443e-01 38.7644 < 2.2e-16 \*\*\*  
## e:KAN 1.4370e+01 8.1369e-01 17.6604 < 2.2e-16 \*\*\*  
## e:KYO 1.3545e+01 1.9818e-01 68.3444 < 2.2e-16 \*\*\*  
## e:NVA 1.3605e+01 5.9121e-01 23.0125 < 2.2e-16 \*\*\*  
## e:OAK 1.2252e+01 2.3988e-01 51.0752 < 2.2e-16 \*\*\*  
## e:SAG 1.3842e+01 3.2668e-01 42.3713 < 2.2e-16 \*\*\*  
## e:SEN 1.4259e+01 2.0668e-01 68.9918 < 2.2e-16 \*\*\*  
## e:WAV 1.3835e+01 2.2243e-01 62.1972 < 2.2e-16 \*\*\*  
## e:PBS 1.2948e+01 4.6632e+00 2.7765 0.0061068 \*\*   
## e:FEM 1.2636e+01 1.4547e+00 8.6864 2.899e-15 \*\*\*  
## e:JAC 1.3716e+01 8.0583e-01 17.0205 < 2.2e-16 \*\*\*  
## e:MAN 1.4025e+01 1.9329e+00 7.2558 1.333e-11 \*\*\*  
## e:NEW 1.4687e+01 7.3310e-01 20.0339 < 2.2e-16 \*\*\*  
## e:NIG 1.4197e+01 NA NA NA   
## e:OKI 1.2741e+01 NA NA NA   
## e:TAK 1.3486e+01 NA NA NA   
## e:YAM 1.3202e+01 NA NA NA   
## e:YAT 1.3538e+01 2.7596e-01 49.0565 < 2.2e-16 \*\*\*  
## e:ZIO 1.3014e+01 7.0766e-02 183.8946 < 2.2e-16 \*\*\*  
## e:FAY 1.3547e+01 3.0099e-01 45.0092 < 2.2e-16 \*\*\*  
## f:BER 9.2641e-01 8.8530e-01 1.0464 0.2968363   
## f:BRU 5.7815e+00 7.6157e+00 0.7592 0.4488031   
## f:KAG 3.0402e+00 3.9448e+00 0.7707 0.4419627   
## f:KAN 3.9370e+00 6.7975e+00 0.5792 0.5632271   
## f:KYO 1.1956e+00 1.2554e+00 0.9524 0.3422622   
## f:NVA 1.0129e+00 2.7457e+00 0.3689 0.7126622   
## f:OAK 3.0568e-01 4.0268e-01 0.7591 0.4488320   
## f:SAG 3.6185e+00 3.4930e+00 1.0359 0.3017037   
## f:SEN 4.0810e+00 4.3820e+00 0.9313 0.3530037   
## f:WAV 2.3921e+00 2.7275e+00 0.8770 0.3817103   
## f:PBS 2.0375e+00 4.6436e+00 0.4388 0.6613828   
## f:FEM 7.3344e+00 2.0275e+02 0.0362 0.9711849   
## f:JAC 5.8284e+00 8.6409e+00 0.6745 0.5009001   
## f:MAN 4.9730e+00 3.7221e+01 0.1336 0.8938710   
## f:NEW 6.5840e+00 1.0316e+01 0.6382 0.5241845   
## f:NIG 3.3452e+00 NA NA NA   
## f:OKI 1.4907e+00 NA NA NA   
## f:TAK 2.4460e-01 NA NA NA   
## f:YAM 1.5278e-01 NA NA NA   
## f:YAT 4.1355e+00 4.1496e+00 0.9966 0.3203711   
## f:ZIO 1.7205e-01 2.1578e-01 0.7973 0.4263594   
## f:FAY 2.6505e+00 3.4205e+00 0.7749 0.4394729   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error:  
##   
## 0.05163599 (171 degrees of freedom)

# estimate the 50% intercept  
ED(d.model.fit.2008, 50)

##   
## Estimated effective doses  
##   
## Estimate Std. Error  
## e:AIZ:50 13.805553 0.093327  
## e:BER:50 13.381520 0.104692  
## e:BRU:50 12.677897 0.043248  
## e:FAY:50 13.089447 0.050913  
## e:FEM:50 12.571691 0.793467  
## e:HIR:50 13.108887 0.094609  
## e:JAC:50 12.675636 0.081978  
## e:KAG:50 13.291741 0.069130  
## e:KHO:50 13.390534 0.065938  
## e:MAN:50 13.386252 0.102768  
## e:NEW:50 13.803045 0.125894  
## e:NVA:50 13.099591 0.071996  
## e:OAK:50 12.312826 0.184102  
## e:OKI:50 12.912827 0.181730  
## e:SAK:50 13.664493 0.064151  
## e:SHI:50 13.122422 0.117793  
## e:TAN:50 12.647793 0.092936  
## e:TOK:50 13.259946 0.010958  
## e:UTS:50 13.588706 0.042961  
## e:WAV:50 13.434880 0.087219  
## e:ZIO:50 13.048402 0.076208

cpp\_values\_2008<-data.frame(ED(d.model.fit.2008, 50))

##   
## Estimated effective doses  
##   
## Estimate Std. Error  
## e:AIZ:50 13.805553 0.093327  
## e:BER:50 13.381520 0.104692  
## e:BRU:50 12.677897 0.043248  
## e:FAY:50 13.089447 0.050913  
## e:FEM:50 12.571691 0.793467  
## e:HIR:50 13.108887 0.094609  
## e:JAC:50 12.675636 0.081978  
## e:KAG:50 13.291741 0.069130  
## e:KHO:50 13.390534 0.065938  
## e:MAN:50 13.386252 0.102768  
## e:NEW:50 13.803045 0.125894  
## e:NVA:50 13.099591 0.071996  
## e:OAK:50 12.312826 0.184102  
## e:OKI:50 12.912827 0.181730  
## e:SAK:50 13.664493 0.064151  
## e:SHI:50 13.122422 0.117793  
## e:TAN:50 12.647793 0.092936  
## e:TOK:50 13.259946 0.010958  
## e:UTS:50 13.588706 0.042961  
## e:WAV:50 13.434880 0.087219  
## e:ZIO:50 13.048402 0.076208

# Combine and format CPP value tables from 1988, 2008 and 2018

cpp\_2018<-data.frame(ED(d.model.fit, 50))

## Warning in sqrt(dEDval %\*% varCov %\*% dEDval): NaNs produced  
  
## Warning in sqrt(dEDval %\*% varCov %\*% dEDval): NaNs produced

##   
## Estimated effective doses  
##   
## Estimate Std. Error  
## e:BER:50 13.600154 0.032426  
## e:BRU:50 12.991086 0.045615  
## e:FAY:50 13.362601 0.049503  
## e:FEM:50 12.517899 0.204107  
## e:JAC:50 12.853244 0.069543  
## e:KAG:50 13.458850 0.032953  
## e:KAN:50 13.764317 0.049295  
## e:KYO:50 13.513563 0.027389  
## e:MAN:50 13.627346 0.040470  
## e:NEW:50 13.844557 0.049230  
## e:NIG:50 13.849662 0.049069  
## e:NVA:50 13.602618 0.043421  
## e:OAK:50 12.514155 0.110016  
## e:OKI:50 12.723369 0.067729  
## e:PBS:50 12.337784 0.978213  
## e:SAG:50 13.432874 0.037531  
## e:SEN:50 14.014586 0.024896  
## e:TAK:50 13.662345 NA  
## e:WAV:50 13.682082 0.024563  
## e:YAM:50 13.527540 0.042665  
## e:YAT:50 13.209568 0.034621  
## e:ZIO:50 13.167965 0.034344

cpp\_2018<-rownames\_to\_column(cpp\_2018, var="pop")   
cpp\_2018$pop<-str\_sub(cpp\_2018$pop,3,5)  
popdata\_2018<-DI\_summary %>% group\_by(pop) %>%   
 summarize(lat=first(lat), country=first(country))  
cpp\_2018<-right\_join(cpp\_2018, popdata\_2018, by="pop")

## Warning: Column `pop` joining character vector and factor, coercing into  
## character vector

cpp\_2018$year<-rep("2018", nrow(cpp\_2018))  
colnames(cpp\_2018)[colnames(cpp\_2018)=="Std..Error"] <- "StdError"  
  
cpp\_2008<-data.frame(ED(d.model.fit.2008, 50)) %>% rownames\_to\_column(var="pop")

##   
## Estimated effective doses  
##   
## Estimate Std. Error  
## e:AIZ:50 13.805553 0.093327  
## e:BER:50 13.381520 0.104692  
## e:BRU:50 12.677897 0.043248  
## e:FAY:50 13.089447 0.050913  
## e:FEM:50 12.571691 0.793467  
## e:HIR:50 13.108887 0.094609  
## e:JAC:50 12.675636 0.081978  
## e:KAG:50 13.291741 0.069130  
## e:KHO:50 13.390534 0.065938  
## e:MAN:50 13.386252 0.102768  
## e:NEW:50 13.803045 0.125894  
## e:NVA:50 13.099591 0.071996  
## e:OAK:50 12.312826 0.184102  
## e:OKI:50 12.912827 0.181730  
## e:SAK:50 13.664493 0.064151  
## e:SHI:50 13.122422 0.117793  
## e:TAN:50 12.647793 0.092936  
## e:TOK:50 13.259946 0.010958  
## e:UTS:50 13.588706 0.042961  
## e:WAV:50 13.434880 0.087219  
## e:ZIO:50 13.048402 0.076208

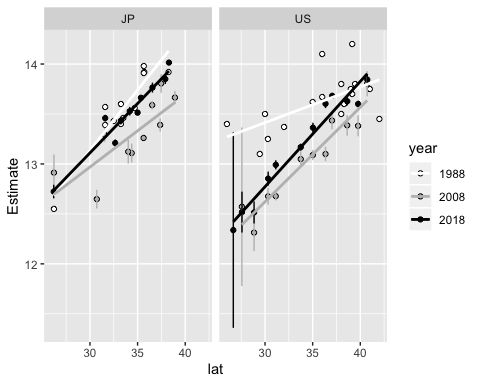
cpp\_2008$pop<-str\_sub(cpp\_2008$pop,3,5)  
colnames(cpp\_2008)[colnames(cpp\_2008)=="Std..Error"] <- "StdError"  
popdata\_2008<-DI\_summary\_2008 %>% group\_by(pop) %>%   
 summarize(country=first(country), lat=first(lat), year=first(year))   
cpp\_2008<-right\_join(cpp\_2008, popdata\_2008, by="pop")

## Warning: Column `pop` joining character vector and factor, coercing into  
## character vector

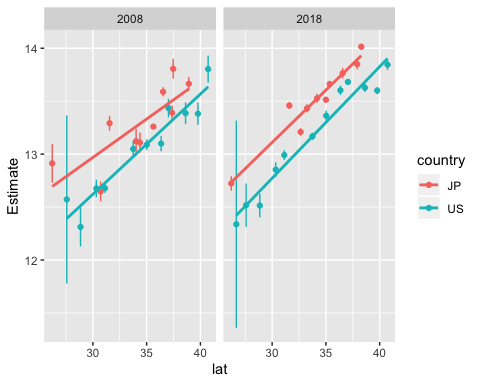
cpp\_1988<-data.frame(read.csv("CPP\_1988.csv"))   
cpp\_1988$pop<-na\_if(cpp\_1988$pop, "NA")  
cpp\_1988$StdError<-na\_if(cpp\_1988$StdError, "NA")  
  
cpp\_2<-rbind(cpp\_1988, cpp\_2008, cpp\_2018)

# CPP by latitude, country and year

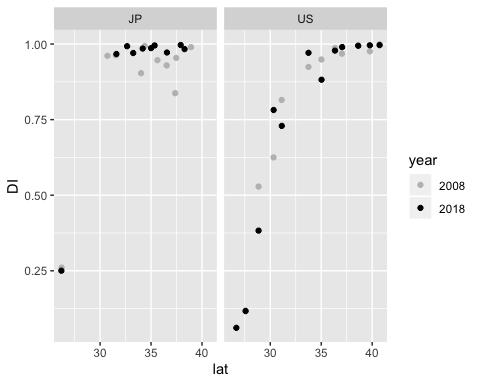
ggplot(cpp\_2, aes(x=lat, y=Estimate, fill=year, color=year)) +   
 scale\_color\_manual(values=c("white","gray","black")) +   
 scale\_fill\_manual(values=c("white","gray","black"))+   
 geom\_point(shape=21, color="black") +   
 geom\_errorbar(aes(ymin=Estimate-StdError, ymax=Estimate+StdError), na.rm=TRUE) +   
 facet\_grid(~country) + stat\_smooth(method=lm, se=FALSE)



ggplot(filter(cpp\_2, year!=1988), aes(x=lat, y=Estimate, color=country)) +   
 geom\_point() +   
 geom\_errorbar(aes(ymin=Estimate-StdError, ymax=Estimate+StdError), na.rm=TRUE) +   
 facet\_grid(~year) + stat\_smooth(method=lm, se=FALSE)

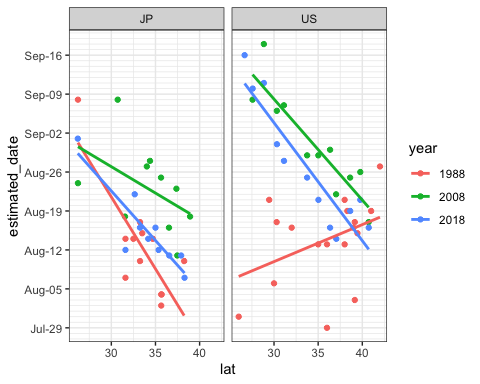


DI\_short\_2008<-filter(DI\_summary\_2008, photoperiod==8)  
  
#unnecessary renaming between pp/photoperiod - harmonize @ beginning?  
DI\_short\_2018<-filter(DI\_summary, pp==8) %>% group\_by(pop) %>%   
 summarize(country=first(country), lat=first(lat), photoperiod=first(pp), emb=sum(emb), total=sum(total))  
DI\_short\_2018$DI<-DI\_short\_2018$emb/DI\_short\_2018$total  
DI\_short\_2018<-DI\_short\_2018[,c(1:4,7)]   
DI\_short\_2018$year<-rep("2018",nrow(DI\_short\_2018))  
  
DI\_short\_2<-rbind(DI\_short\_2008, DI\_short\_2018)  
write\_csv(DI\_short\_2, "diapause\_incidence\_8h\_2008\_2018.csv")  
  
ggplot(DI\_short\_2, aes(x=lat, y=DI, color=year, label=pop)) + scale\_color\_manual(values=c("gray","black")) +   
 geom\_point() + facet\_grid(~country)



findDate<-function(lat, pp, year) {  
#generate day lengths at latitude between July 1 and Dec 31  
x<-daylength(lat,182:365)  
#find position of day length in the daylength vector that is closest to photoperiod queried  
daynumber<-which.min(abs(x-pp))  
#convert day number to date  
origin<-as.character(paste(year,"-07-01",sep=""))  
return(as.character(as.Date(daynumber, origin=origin)))  
}

cpp\_2$date<-mapply(findDate, lat=cpp\_2$lat, pp=cpp\_2$Estimate, year=cpp\_2$year)  
cpp\_2$estimated\_date<-format(as.Date(cpp\_2$date), "%mm-%dd")  
cpp\_2$estimated\_date<-as.Date(cpp\_2$estimated\_date, format="%mm-%dd")  
ggplot(cpp\_2, aes(x=lat, y=estimated\_date, color=year)) + geom\_point() + scale\_y\_date(date\_breaks = "weeks" , date\_minor\_breaks="days", date\_labels = "%b-%d") + facet\_grid(~country) + theme\_bw() + stat\_smooth(method=lm, se=FALSE)



The approximate date of 50% diapause in Berlin, NJ in 2018 was 2019-08-21.