doi:10.1038/nature20150

Contents

List of Tables

LMER model selection

There were several LMER models which were considered as follows:

```
l_ply(names(lmer.list), function(xx){
  cat('------------',xx,'---------------\n')
  print(summary(lmer.list[[xx]]))
  cat('\n')})
```

```
## ------------ simple ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control + (1 | Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 298.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5710 -0.4376  0.0100  0.5099  3.3646
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Study    (Intercept) 0.004785 0.06917
```

```
##  Residual             0.224720 0.47405
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.12979    0.04796   2.706
## C.control    0.86890    0.03401  25.548
##
## Correlation of Fixed Effects:
##           (Intr)
## C.control -0.701
##
## ------------ addative.dT ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control + Tdelta + (1 | Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 302.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5657 -0.4340  0.0158  0.5067  3.3363
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  Study    (Intercept) 0.0008106 0.02847
##  Residual             0.2281379 0.47764
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.15543    0.05478   2.837
## C.control    0.88241    0.03342  26.406
## Tdelta      -0.03801    0.03348  -1.135
##
## Correlation of Fixed Effects:
##           (Intr) C.cntr
## C.control -0.519
## Tdelta    -0.523 -0.146
##
## ------------ addative.all ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control + MAP + MAT + pH + degYr + perClay + (1 |
##     Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 310.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.3695 -0.5154  0.0043  0.4686  3.5373
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
```

```
##  Study    (Intercept) 0.004465 0.06682
##  Residual             0.219005 0.46798
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.08816    0.07787   1.132
## C.control    0.83458    0.03868  21.576
## MAP          0.14869    0.06882   2.161
## MAT         -0.16039    0.06317  -2.539
## pH           0.11411    0.05062   2.254
## degYr       -0.06862    0.03717  -1.846
## perClay      0.04239    0.03852   1.100
##
## Correlation of Fixed Effects:
##           (Intr) C.cntr MAP    MAT    pH     degYr
## C.control -0.384
## MAP       -0.247 -0.444
## MAT       -0.004  0.456 -0.801
## pH        -0.365 -0.261  0.693 -0.545
## degYr     -0.143  0.040 -0.272  0.140 -0.399
## perClay   -0.199  0.131 -0.177 -0.074 -0.269  0.063
##
## ------------ addative.enviro ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control + MAP + MAT + pH + perClay + (1 | Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 309.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.3981 -0.4809  0.0177  0.5070  3.5172
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Study    (Intercept) 0.01051  0.1025
##  Residual             0.21674  0.4656
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.07561    0.08078   0.936
## C.control    0.82323    0.04028  20.438
## MAP          0.12463    0.06944   1.795
## MAT         -0.15238    0.06553  -2.325
## pH           0.08068    0.04869   1.657
## perClay      0.04554    0.04062   1.121
##
## Correlation of Fixed Effects:
##           (Intr) C.cntr MAP    MAT    pH
## C.control -0.374
## MAP       -0.305 -0.446
## MAT        0.019  0.449 -0.798
```

```
## pH        -0.466 -0.267  0.661 -0.537
## perClay   -0.197  0.131 -0.168 -0.083 -0.265
##
## ------------ addative.treat ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control + degYr + (1 | Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 302.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5364 -0.4377 -0.0072  0.4943  3.3422
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Study    (Intercept) 0.00174  0.04171
##  Residual             0.22723  0.47669
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.16115    0.05703   2.826
## C.control    0.87610    0.03330  26.313
## degYr       -0.03774    0.03330  -1.133
##
## Correlation of Fixed Effects:
##           (Intr) C.cntr
## C.control -0.564
## degYr     -0.567 -0.030
##
## ------------ interactive ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control * degYr + (1 | Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 301.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5792 -0.4282 -0.0107  0.4858  3.3985
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Study    (Intercept) 0.006856 0.0828
##  Residual             0.217700 0.4666
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##                 Estimate Std. Error t value
## (Intercept)      0.10596    0.06369   1.664
## C.control        0.94370    0.04690  20.120
## degYr            0.03422    0.04482   0.763
## C.control:degYr -0.08237    0.03407  -2.418
```

```
##
## Correlation of Fixed Effects:
##              (Intr) C.cntr degYr
## C.control    -0.645
## degYr        -0.655  0.421
## C.cntrl:dgY   0.404 -0.686 -0.646
##
## ------------ interactive.dT ---------------
## Linear mixed model fit by REML ['lmerMod']
## Formula: C.warmed ~ C.control * Tdelta + (1 | Study)
##    Data: data.sample.plus.rescaled
##
## REML criterion at convergence: 304.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.7097 -0.4567  0.0184  0.4999  3.3500
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Study    (Intercept) 0.001885 0.04342
##  Residual             0.224830 0.47416
## Number of obs: 213, groups:  Study, 47
##
## Fixed effects:
##                   Estimate Std. Error t value
## (Intercept)        0.08639    0.06785   1.273
## C.control          0.93133    0.04422  21.063
## Tdelta             0.04506    0.05736   0.785
## C.control:Tdelta -0.05523    0.03109  -1.776
##
## Correlation of Fixed Effects:
##              (Intr) C.cntr Tdelta
## C.control    -0.699
## Tdelta       -0.724  0.463
## C.cntrl:Tdl   0.585 -0.652 -0.810
```

Comparing the BIC scores between the models, the simple regression between the carbon stock in the warmed plots and the mean carbon stock of the control plots has the best score. The model with the addative degree-years or degrees preforms best if we want more then just the basic correlation. There is no notable difference between degree-years and degrees as a determinent for warmed soil carbon stocks.

```
pander(anova(lmer.list$simple, lmer.list$addative.treat,
             lmer.list$addative.dT, lmer.list$addative.enviro,
             lmer.list$addative.all), caption='Model fits comparing the statistical power
       gained by of treatment (degree-Years, and degree; addative.treat and
       addative.dT respectively) vs enviromental variables (MAT, MAP, and pH;
       addative.enviro) vs all variables include (addative.enviro) to
       explaining warmed soil carbon stocks.')
```

```
## refitting model(s) with ML (instead of REML)
```

Table 1: Model fits comparing the statistical power gained by of treatment (degree-Years, and degree; addative.treat and addative.dT respectively) vs enviromental variables (MAT, MAP, and pH; addative.enviro) vs all variables include (addative.enviro) to explaining warmed soil carbon stocks.

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
lmer.list$simple	4	296.7	310.2	-144.4	288.7	NA	NA	NA
lmer.list$addative.treat	5	297.4	314.2	-143.7	287.4	1.368	1	0.2422
lmer.list$addative.dT	5	297.4	314.2	-143.7	287.4	0	0	1
lmer.list$addative.enviro	8	297.3	324.2	-140.7	281.3	6.048	3	0.1093
lmer.list$addative.all	9	295.6	325.8	-138.8	277.6	3.785	1	0.0517

The interactive model has both a better AIC and BIC score then even the simple regression. Thus the interative model is the most parsimonious.

```
pander(anova(lmer.list$interactive, lmer.list$interactive.dT, lmer.list$addative.treat,
              lmer.list$simple),
       caption='Model fits comparing the statistical power gained by multiplicative
       vs addative models using the controlled soil carbon stocks and degree-years or degrees
       warmed to explain warmed soil carbon stocks. The interactive degree-years model
       (interactive) signficantly better then the alternative models
       (interactive.dT, addative.treat, and simple) considered.')
```

```
## refitting model(s) with ML (instead of REML)
```

Table 2: Model fits comparing the statistical power gained by multiplicative vs addative models using the controlled soil carbon stocks and degree-years or degrees warmed to explain warmed soil carbon stocks. The interactive degree-years model (interactive) signficantly better then the alternative models (interactive.dT, addative.treat, and simple) considered.

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
lmer.list$simple	4	296.7	310.2	-144.4	288.7	NA	NA	NA
lmer.list$addative.treat	5	297.4	314.2	-143.7	287.4	1.368	1	0.2422
lmer.list$interactive	6	293.7	313.9	-140.8	281.7	5.665	1	0.01731
lmer.list$interactive.dT	6	296.2	316.3	-142.1	284.2	0	0	1

doi:10.1038/nature20150

Linear regression models

```
pander(merge(subset(modelFits, data=='data.sample', select=-data),
             subset(modelFits, data=='data.study', select=-data),
             by=c('model'), suffixes=c('.sample', '.study'))[,c('model', 'adjR2.sample', 'pvalue.sample
       caption='R2 and p-value of the control soil carbon stock and degree-years or degrees to
       explain warmed soil carbon stocks, the difference between warmed and control soil carbon
       stocks, and the rate of change of soil carbon stocks per degree-year across samples and
       studies.')
```

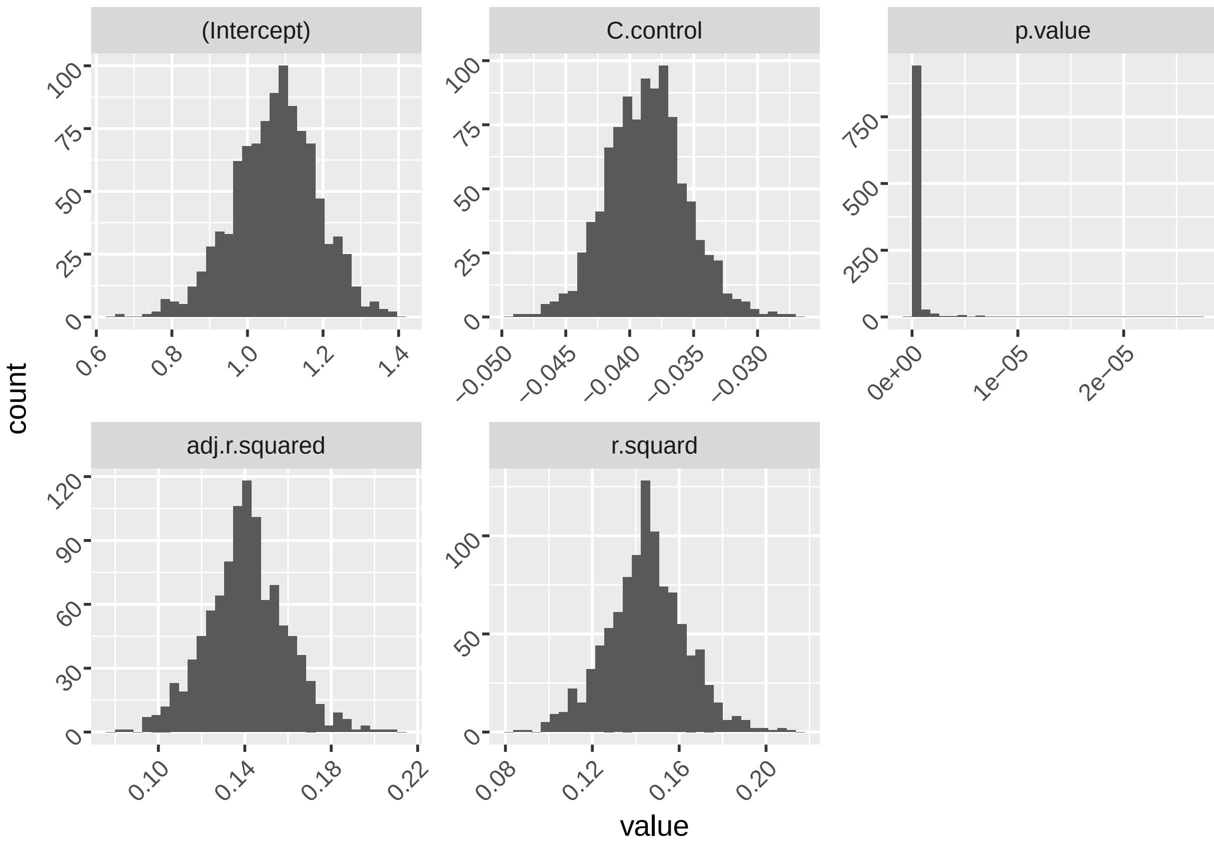
Table 3: R2 and p-value of the control soil carbon stock and degree-years or degrees to explain warmed soil carbon stocks, the difference between warmed and control soil carbon stocks, and the rate of change of soil carbon stocks per degree-year across samples and studies.

model	adjR2.sample	pvalue.sample	adjR2.study	pvalue.study
(C.warmed - C.control)/(Years * Tdelta) ~ C.control	0.151	2.09e-09	0.489	1.4e-08
(C.warmed - C.control)/Tdelta ~ C.control	0.145	4.52e-09	0.304	2.37e-05
C.warmed - C.control ~ C.control * degYr	0.442	3.13e-27	0.606	8.22e-10
C.warmed - C.control ~ C.control * Tdelta	0.411	8.82e-25	0.529	4.32e-08
C.warmed ~ C.control * degYr	0.815	9.59e-78	0.953	1.36e-30
C.warmed ~ C.control * Tdelta	0.805	2.78e-75	0.944	7.51e-29

CI for parameter range

```
ggplot(melt(dCperDegYr.boot)) +
  geom_histogram(aes(x=value)) + facet_wrap(~variable, scales='free') +
  theme(axis.text=element_text(angle = 45, hjust = 1))
```

```
## No id variables; using all as measure variables
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
pander(subset(parRange, type %in% resultsTable$type), caption='95%CI of the coefficents and R2 of the c
       (warmed-controlled) per degree-year explained by the control soil carbon stock
       [kg-C m^-3], constructed from samples. The type key is as follows: dCperDegYr is the
       change in carbon stock regressed against the degree-year, dCperDeg is the change in
       carbon stock regressed against the degrees warmed, and the time notates a dC per
       degree-year regression where study times were capped at the stated time (ie for yr1 any
       study that ran longer then a year was set to one year and then the change in carbon
       stock agaist degree-year was calculated). ')
```

Table 4: 95%CI of the coefficents and R2 of the change in soil carbon stocks (warmed-controlled) per degree-year explained by the control soil carbon stock [kg-C m^-3], constructed from samples. The type key is as follows: dCperDegYr is the change in carbon stock regressed against the degree-year, dCperDeg is the change in carbon stock regressed against the degrees warmed, and the time notates a dC per degree-year regression where study times were capped at the stated time (ie for yr1 any study that ran longer then a year was set to one year and then the change in carbon stock agaist degree-year was calculated).

	type	intercept	C	p.value	adj.r.squared	r.squard	qrt
1	dCperDegYr	0.8866	-0.04358	9.34e-10	0.1089	0.1134	0.05
2	dCperDegYr	1.076	-0.03864	3.083e-08	0.1407	0.145	0.5

	type	intercept	C	p.value	adj.r.squared	r.squard	qrt
3	dCperDegYr	1.254	-0.03334	1.188e-06	0.1701	0.1743	0.95
4	dCperDeg	4.001	-0.2091	2.298e-09	0.09723	0.1018	0.05
5	dCperDeg	5.399	-0.1855	9.812e-08	0.1308	0.1352	0.5
6	dCperDeg	6.223	-0.1319	4.448e-06	0.1627	0.1669	0.95
7	wk1	198.6	-10.94	2.241e-09	0.09451	0.09911	0.05
8	wk1	280.4	-9.688	1.216e-07	0.129	0.1334	0.5
9	wk1	323.8	-6.559	6.267e-06	0.1626	0.1668	0.95
10	mon1	46.13	-2.52	1.473e-09	0.09606	0.1006	0.05
11	mon1	64.59	-2.237	1.03e-07	0.1304	0.1348	0.5
12	mon1	74.9	-1.515	5.263e-06	0.1664	0.1706	0.95
13	mon6	7.253	-0.416	1.98e-09	0.08499	0.08964	0.05
14	mon6	10.72	-0.3699	1.025e-07	0.1303	0.1347	0.5
15	mon6	12.34	-0.2427	1.829e-05	0.1633	0.1675	0.95
16	yr1	3.901	-0.2117	1.483e-09	0.09832	0.1029	0.05
17	yr1	5.5	-0.1893	7.171e-08	0.1335	0.1379	0.5
18	yr1	6.268	-0.1288	4.002e-06	0.1659	0.1701	0.95
22	yr5	1.16	-0.0577	2.853e-10	0.1218	0.1262	0.05
23	yr5	1.451	-0.05142	9.718e-09	0.1503	0.1546	0.5
24	yr5	1.688	-0.04257	2.814e-07	0.1799	0.1841	0.95
25	yr7	1.033	-0.05116	1.995e-10	0.1197	0.1241	0.05
26	yr7	1.264	-0.0455	5.731e-09	0.155	0.1593	0.5
27	yr7	1.474	-0.03836	3.57e-07	0.1823	0.1864	0.95
31	yr8.75	0.9666	-0.04752	2.73e-10	0.1166	0.1211	0.05
32	yr8.75	1.169	-0.04227	8.245e-09	0.1518	0.1561	0.5
33	yr8.75	1.348	-0.03643	4.96e-07	0.1803	0.1844	0.95
37	yr11.6	0.9024	-0.04494	5.163e-10	0.1146	0.1191	0.05
38	yr11.6	1.096	-0.03992	1.592e-08	0.1463	0.1506	0.5

doi:10.1038/nature20150

	type	intercept	C	p.value	adj.r.squared	r.squard	qrt
39	yr11.6	1.281	-0.03461	6.246e-07	0.175	0.1792	0.95
43	yr17.5	0.8823	-0.04337	9.387e-10	0.1086	0.1132	0.05
44	yr17.5	1.067	-0.03851	3.577e-08	0.1394	0.1438	0.5
45	yr17.5	1.246	-0.03345	1.263e-06	0.1701	0.1743	0.95
55	yr35	0.8928	-0.04337	9.072e-10	0.1112	0.1158	0.05
56	yr35	1.071	-0.03878	3.3e-08	0.14	0.1444	0.5
57	yr35	1.243	-0.03366	9.32e-07	0.1702	0.1744	0.95

Global Extrapolations

```
temp <- subset(resultsTable, globalWarming %in% c(1,2), c('type', 'globalWarming', 'warmingDistribution
row.names(temp) <- NULL
pander(temp,
       caption='Global soil carbon change across effect-time assumptions. Type is analygous
       to the key described above. Global warming is the average global warming applied
       linearlly over 35 years. Time step is the size of the time step used in the numerical
       integration. dC is the change in the soil carbon stock for the 5% quantile, 50% quantile,
       and 95% quantile respectively calcuated form the parameter ranges described above.',
       round=c(1,1,1,3,0,0,0))
```

Table 5: Global soil carbon change across effect-time assumptions. Type is analygous to the key described above. Global warming is the average global warming applied linearlly over 35 years. Time step is the size of the time step used in the numerical integration. dC is the change in the soil carbon stock for the 5% quantile, 50% quantile, and 95% quantile respectively calcuated form the parameter ranges described above.

type	globalWarming	warmingDistribution	timeStep	dC_qrt05	dC_qrt50	dC_qrt95
dCperDeg	1	unif	NA	-103	-55	9
dCperDegYr	1	unif	0.019	0	0	0
dCperDegYr	1	unif	0.083	0	0	0
dCperDegYr	1	unif	0.5	0	0	0
dCperDegYr	1	unif	1	0	0	0
dCperDegYr	1	unif	10	-29	-18	-6
dCperDegYr	1	unif	11.67	-40	-24	-8

doi:10.1038/nature20150

type	globalWarming	warmingDistribution	timeStep	dC_qrt05	dC_qrt50	dC_qrt95
dCperDegYr	1	unif	17.5	-89	-54	-18
dCperDegYr	1	unif	20	-117	-70	-24
dCperDegYr	1	unif	25	-183	-110	-37
dCperDegYr	1	unif	30	-263	-158	-53
dCperDegYr	1	unif	35	-358	-216	-72
dCperDegYr	1	unif	4	-5	-3	-1
dCperDegYr	1	unif	5	-7	-4	-1
dCperDegYr	1	unif	7	-14	-9	-3
dCperDegYr	1	unif	8	-19	-11	-4
dCperDegYr	1	unif	8.75	-22	-13	-5
mon1	1	unif	0.083	-50	-26	7
mon6	1	unif	0.5	-51	-26	8
wk1	1	unif	0.019	-50	-26	7
yr1	1	unif	1	-51	-27	6
yr11.6	1	unif	11.67	-114	-70	-26
yr17.5	1	unif	17.5	-161	-99	-36
yr35	1	unif	35	-352	-219	-81
yr5	1	unif	5	-64	-38	-8
yr7	1	unif	7	-79	-48	-14
yr8.75	1	unif	8.75	-90	-56	-20
dCperDeg	1	CESM	NA	-103	-55	9
dCperDegYr	1	CESM	0.019	0	0	0
dCperDegYr	1	CESM	0.083	0	0	0
dCperDegYr	1	CESM	0.5	0	0	0
dCperDegYr	1	CESM	1	0	0	0
dCperDegYr	1	CESM	10	-29	-18	-6
dCperDegYr	1	CESM	11.67	-40	-24	-8
dCperDegYr	1	CESM	17.5	-89	-54	-18
dCperDegYr	1	CESM	20	-116	-70	-24
dCperDegYr	1	CESM	25	-177	-107	-35
dCperDegYr	1	CESM	30	-249	-149	-48

type	globalWarming	warmingDistribution	timeStep	dC_qrt05	dC_qrt50	dC_qrt95
dCperDegYr	1	CESM	35	-326	-195	-60
dCperDegYr	1	CESM	4	-5	-3	-1
dCperDegYr	1	CESM	5	-7	-4	-2
dCperDegYr	1	CESM	7	-14	-9	-3
dCperDegYr	1	CESM	8	-19	-11	-4
dCperDegYr	1	CESM	8.75	-22	-14	-5
mon1	1	CESM	0.083	-49	-25	7
mon6	1	CESM	0.5	-51	-25	8
wk1	1	CESM	0.019	-49	-25	7
yr1	1	CESM	1	-50	-26	6
yr11.6	1	CESM	11.67	-110	-68	-24
yr17.5	1	CESM	17.5	-153	-94	-33
yr35	1	CESM	35	-321	-199	-68
yr5	1	CESM	5	-63	-37	-8
yr7	1	CESM	7	-77	-46	-13
yr8.75	1	CESM	8.75	-88	-54	-19
dCperDeg	2	unif	NA	-206	-109	18
dCperDegYr	2	unif	0.019	0	0	0
dCperDegYr	2	unif	0.083	0	0	0
dCperDegYr	2	unif	0.5	0	0	0
dCperDegYr	2	unif	1	-1	0	0
dCperDegYr	2	unif	10	-58	-35	-12
dCperDegYr	2	unif	11.67	-80	-48	-16
dCperDegYr	2	unif	17.5	-179	-108	-36
dCperDegYr	2	unif	20	-234	-141	-47
dCperDegYr	2	unif	25	-365	-220	-74
dCperDegYr	2	unif	30	-524	-317	-106
dCperDegYr	2	unif	35	-569	-397	-144
dCperDegYr	2	unif	4	-9	-6	-2
dCperDegYr	2	unif	5	-15	-9	-3
dCperDegYr	2	unif	7	-29	-17	-6

type	globalWarming	warmingDistribution	timeStep	dC_qrt05	dC_qrt50	dC_qrt95
dCperDegYr	2	unif	8	-37	-23	-8
dCperDegYr	2	unif	8.75	-45	-27	-9
mon1	2	unif	0.083	-94	-50	13
mon6	2	unif	0.5	-98	-49	16
wk1	2	unif	0.019	-95	-49	13
yr1	2	unif	1	-97	-51	12
yr11.6	2	unif	11.67	-208	-130	-48
yr17.5	2	unif	17.5	-289	-180	-66
yr35	2	unif	35	-564	-402	-161
yr5	2	unif	5	-121	-72	-16
yr7	2	unif	7	-147	-90	-26
yr8.75	2	unif	8.75	-167	-104	-39
dCperDeg	2	CESM	NA	-200	-105	18
dCperDegYr	2	CESM	0.019	0	0	0
dCperDegYr	2	CESM	0.083	0	0	0
dCperDegYr	2	CESM	0.5	0	0	0
dCperDegYr	2	CESM	1	-1	0	0
dCperDegYr	2	CESM	10	-59	-35	-12
dCperDegYr	2	CESM	11.67	-80	-48	-16
dCperDegYr	2	CESM	17.5	-174	-105	-35
dCperDegYr	2	CESM	20	-223	-134	-43
dCperDegYr	2	CESM	25	-332	-198	-61
dCperDegYr	2	CESM	30	-428	-261	-74
dCperDegYr	2	CESM	35	-478	-291	-74
dCperDegYr	2	CESM	4	-9	-6	-2
dCperDegYr	2	CESM	5	-15	-9	-3
dCperDegYr	2	CESM	7	-29	-17	-6
dCperDegYr	2	CESM	8	-38	-23	-8
dCperDegYr	2	CESM	8.75	-45	-27	-9
mon1	2	CESM	0.083	-91	-48	13
mon6	2	CESM	0.5	-95	-47	16
wk1	2	CESM	0.019	-92	-47	13
yr1	2	CESM	1	-94	-49	12

doi:10.1038/nature20150

type	globalWarming	warmingDistribution	timeStep	dC_qrt05	dC_qrt50	dC_qrt95
yr11.6	2	CESM	11.67	-194	-120	-42
yr17.5	2	CESM	17.5	-255	-157	-52
yr35	2	CESM	35	-472	-297	-88
yr5	2	CESM	5	-116	-68	-14
yr7	2	CESM	7	-139	-85	-23
yr8.75	2	CESM	8.75	-157	-97	-35

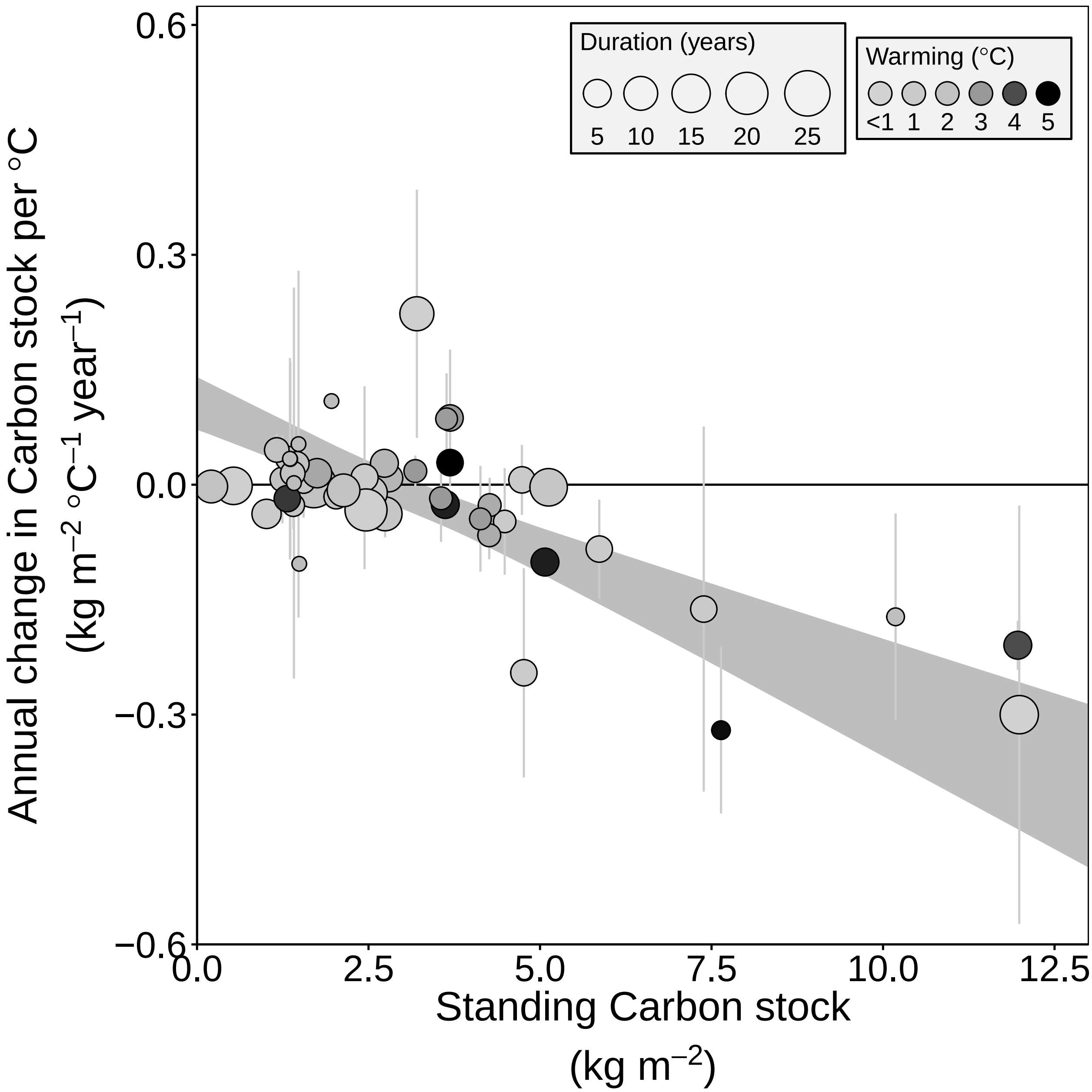
Figures

Change in carbon per degree year with bootstrap (Figure 1)

```
Fig1.theme <- theme(axis.text.x=element_text(size=18,angle=0,colour="black"),
                    axis.text.y=element_text(size=18,angle=0,colour="black"),
                    axis.title=element_text(size=20),
                    legend.text=element_text(size=12),
                    axis.line.x=element_line(color="black"),
                    legend.position = "top",
                    legend.key = element_rect(fill="grey95",size=0,color="grey95"),
                    legend.key.size = unit(0.1,"cm"),
                    legend.title = element_text(size=12,face="bold"),
                    legend.background = element_rect(fill="grey95",color="black"),
                    axis.line = element_line(colour = "black"),
                    panel.grid.major = element_blank(),
                    panel.grid.minor = element_blank(),
                    strip.background = element_rect(colour = "black",size = 0.5),
                    panel.background = element_rect(colour="black", fill="white"),
                    panel.border = element_blank(),
                    axis.ticks = element_line(colour="black"),
                    legend.box = "horizontal",
                    axis.title.y=element_text(vjust=1.9),
                    axis.title.x=element_text(vjust=-0.4))+
                    theme(legend.justification=c(1,1),
                    legend.position=c(1,1))

  # set color gradient
#ramp <- colorRamp(c("black","darkred","red"))
ramp <- colorRamp(c('lightgrey', 'grey', 'black'))
use.col.points <- c(rgb( ramp(seq(0, 1, length = 500)), max = 255))

# generate figure 1
fieldDepth <- 0.1
Figure1 <- ggplot(data.study,aes(x=C.control*fieldDepth, y=dC.perDegYr*fieldDepth)) +
  geom_abline(aes(intercept=parBins$intercept*fieldDepth, slope=parBins$slope),
              colour="grey",data=parBins) +
  geom_abline(intercept=0,slope=0,color="black") +
  geom_errorbar(aes(ymax=(dC.perDegYr + dC.perDegYr.se)*fieldDepth,
                    ymin=(dC.perDegYr - dC.perDegYr.se)*fieldDepth),
                width=0,color="grey80",size=0.5) +
```

```
  geom_point(alpha=1, aes(fill=Tdelta, size=Years), color='black', pch=21)  +
  scale_fill_gradientn(limits=range(c(0,data.study$Tdelta)),
                       colours=use.col.points, space="Lab",labels=c("<1",1,2,3,4,5))+
  scale_size(range=c(3,10)) +
  xlab(expression(atop("Standing Carbon stock","(kg m"^-2*")"))) +
  ylab(expression(atop("Annual change in Carbon stock per"*~degree* C,
                      "(kg m"^-2~degree*C^-1~year^-1*")"))) +
  scale_x_continuous(limits=c(0,0.130*1e3)*fieldDepth, expand = c(0, 0)) +
  scale_y_continuous(limits=c(-6,6.25)*fieldDepth, expand = c(0, 0)) +
  geom_hline(yintercept=6.25*fieldDepth) +
  geom_vline(xintercept=130*fieldDepth) +
  guides(fill = guide_legend(by.row=T,nrow = 1, label.position = "bottom",
                             label.hjust=0.5,title.position="top",
                             title=expression("Warming ("*degree*C*")"),
                             override.aes = list(size = 5),legend.box = "vertical"))+
  guides(size = guide_legend(nrow = 1,label.position = "bottom",
                            label.hjust=0.5,title.position="top",
                            title=expression("Duration (years)"),
                            legend.box = "vertical")) +
  Fig1.theme

print(Figure1)
```

doi:10.1038/nature20150




```
ggsave(plot = Figure1,
       filename='../figs/Figure01.pdf', width=7.5, height=7.5, useDingbats=FALSE)
```

Model-data plot for interactive statisitcal model (Figure 2a)

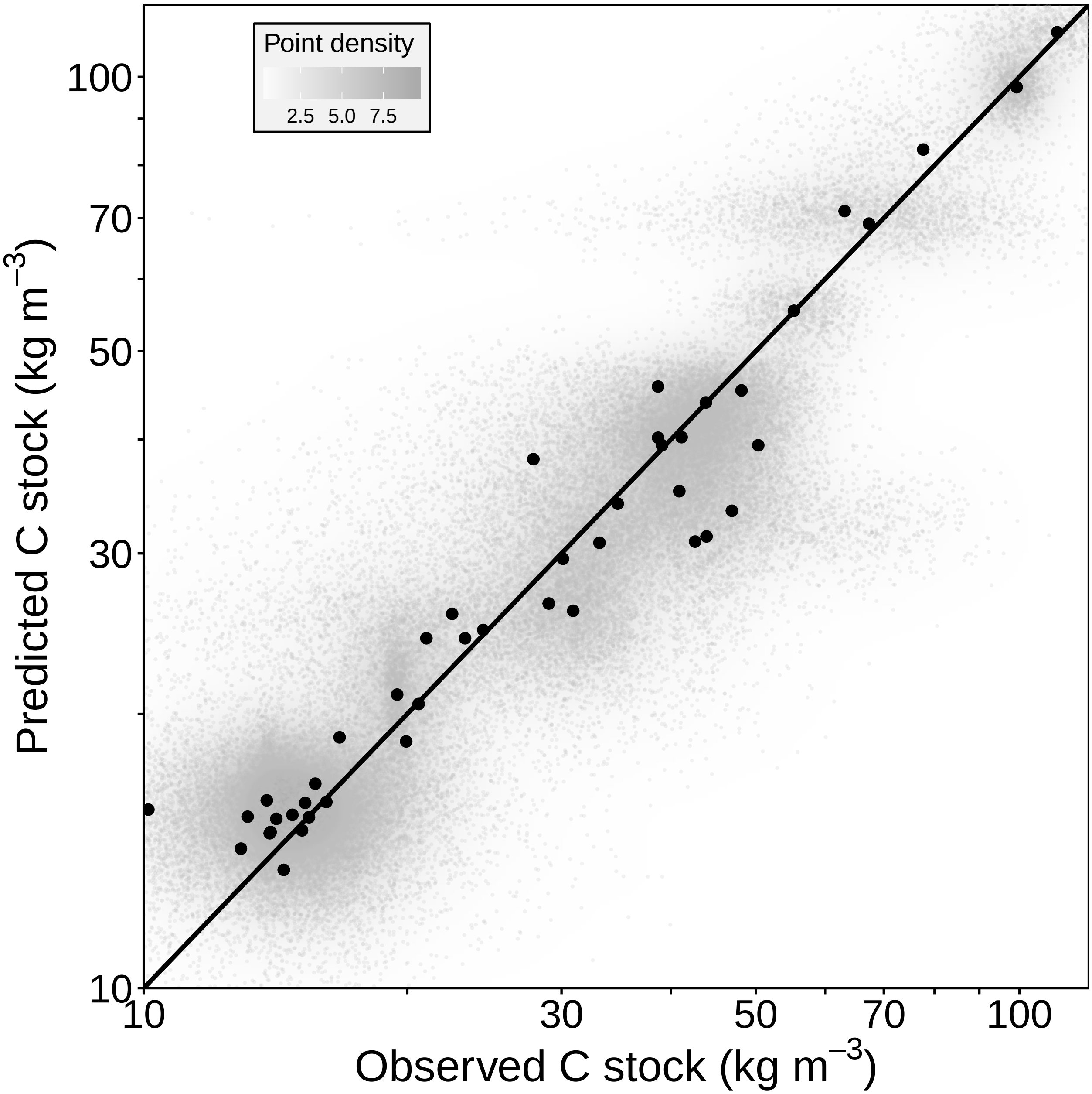
```
print(summary(lm.list$Cw.study))
```

```
##
## Call:
## lm(formula = C.warmed ~ C.control * degYr, data = data.study)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.3269  -2.1202  -0.5347   0.8649  14.0377
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.61834    1.56580   1.034    0.307
## C.control        0.96045    0.03789  25.350  < 2e-16 ***
## degYr            0.30065    0.12352   2.434    0.019 *
## C.control:degYr -0.01662    0.00321  -5.176 5.11e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.924 on 45 degrees of freedom
## Multiple R-squared:  0.9563, Adjusted R-squared:  0.9534
## F-statistic: 328.4 on 3 and 45 DF,  p-value: < 2.2e-16
```

```
#ramp <- colorRamp(c("white","blue","gold","orange","red"))
ramp <- colorRamp(c('white', 'darkgrey'))
use.fill <- rgb( ramp(seq(0, 1, length = 255)), max = 255)
fig2aTheme <- theme(axis.text.x=element_text(size=18,angle=0,colour="black"),
                    axis.text.y=element_text(size=18,angle=0,colour="black"),
                    axis.title=element_text(size=20),
                    axis.line = element_line(colour = "black"),
                    panel.grid.major = element_blank(),
                    panel.grid.minor = element_blank(),
                    strip.background = element_rect(colour = "black",size = 0.5),
                    panel.background = element_rect(colour="black", fill="white"),
                    panel.border = element_blank(),axis.ticks = element_line(colour="black"),
                    legend.box = "vertical",
                    legend.justification=c(0.9,1), legend.position=c(0.3,1),
                    legend.key = element_rect(fill="grey95",size=0,color="grey95"),
                    legend.key.size = unit(0.5,"cm"),
                    legend.title = element_text(size=12,face="bold"),
                    legend.background = element_rect(fill="grey95",color="black"))

figure2a <- ggplot(modelData.df,aes(x=rnd.data,y=rnd.model)) +
  stat_density2d(geom = "raster",aes(fill = ..density..), contour = FALSE,
                 interpolate = TRUE,n=200,show.legend=T) +
  geom_point(size=0.15,alpha=0.2,col="grey") +
  geom_point(data=summaryMD.df,aes(x=data.mean, y=model.mean),
             color="black", size=2) +
  scale_fill_gradientn(colours = use.fill) +
  geom_abline(intercept=0,slope=1,size=1)+
  scale_x_log10(limits=c(10,0.12*1e3), expand = c(0, 0),
                breaks=c(1:10)*10,labels=c(10,"",30,"",50,"",70,"","",100)) +
  scale_y_log10(limits=c(10,0.12*1e3), expand = c(0, 0),
                breaks=c(1:10)*10,labels=c(10,"",30,"",50,"",70,"","",100)) +
  xlab(bquote("Observed C stock (kg "*m^-3*")")) +
  ylab(bquote("Predicted C stock (kg "*m^-3*")")) +
  guides( fill = guide_colourbar(label.position = "bottom",
                                 label.hjust=0.5,title.position="top",
                                 title=expression("Point density"), direction = "horizontal")) +
  fig2aTheme
```

```
print(figure2a)
```



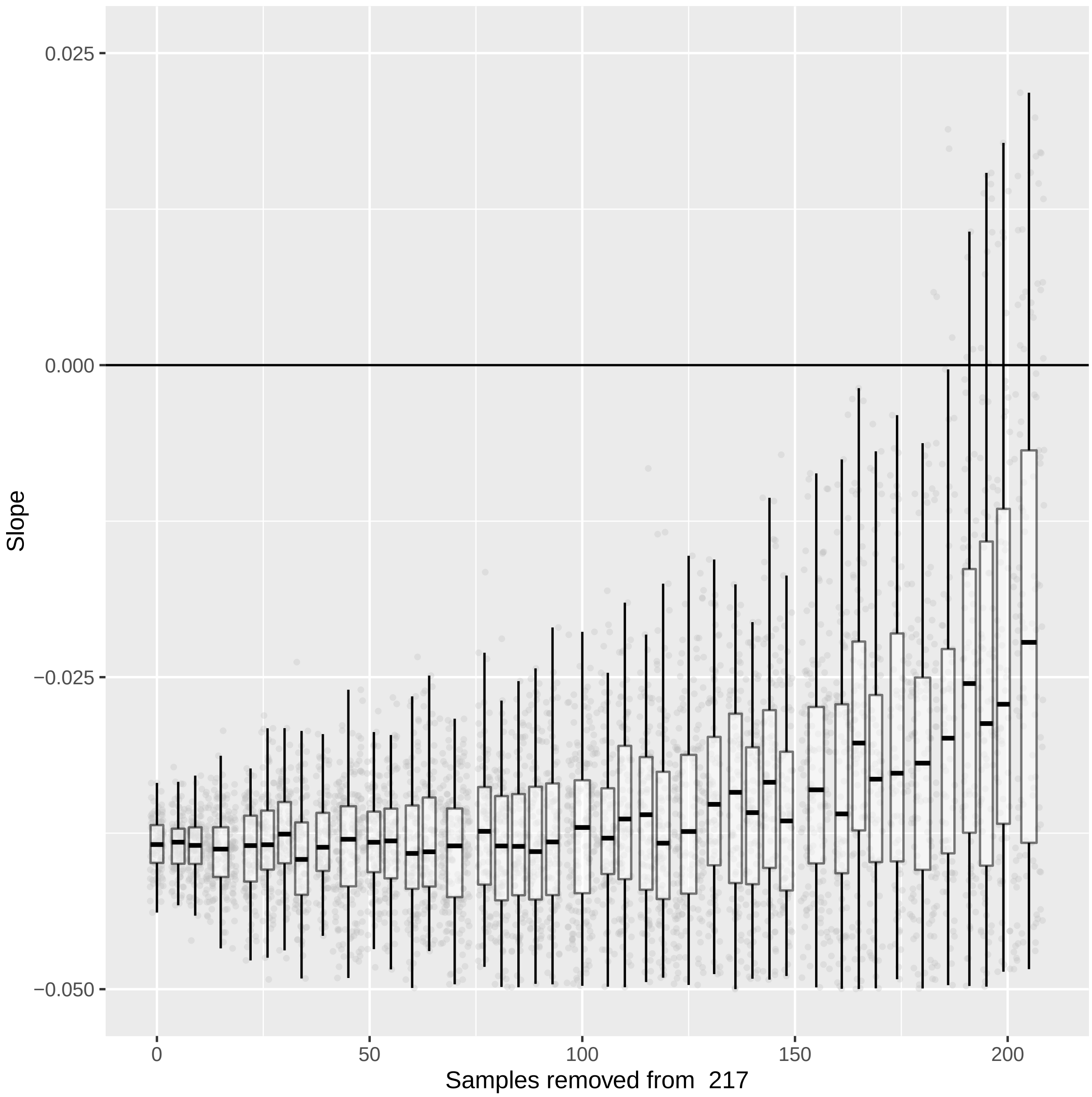

```
ggsave(plot=figure2a,
       file='../figs/Figure02a.pdf', height=7, width=7, useDingbats=FALSE)
```

Boot strap slope comparison (Figure 2b)

```
ggplot(selectSize.sample, aes(x=dim(data.sample)[1]-sampleSize, y=C.control)) +
  geom_jitter(alpha=0.3,color="grey",height=0,size=0.75) +
            scale_y_continuous(limits = c(-0.05, 0.025)) +
```

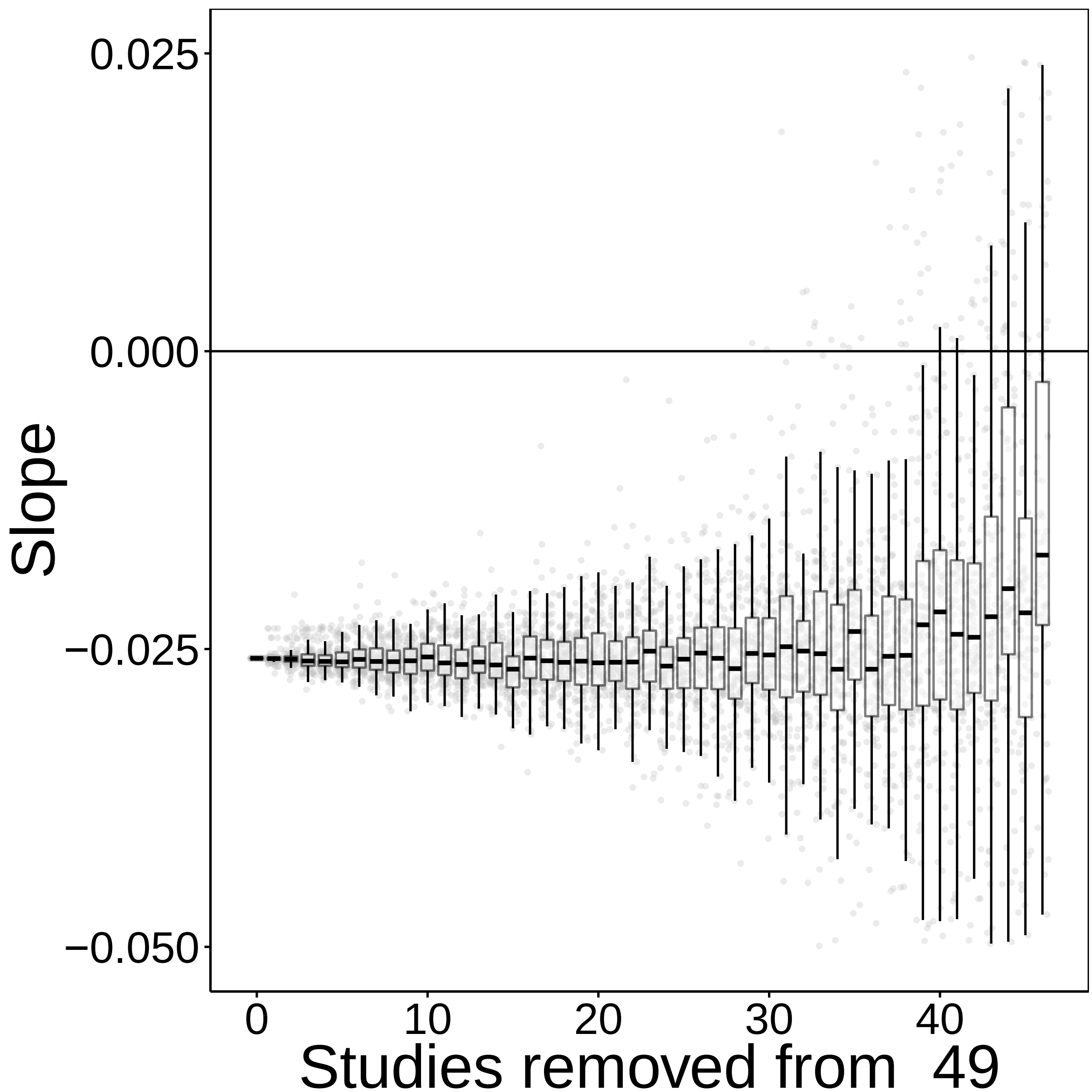
doi:10.1038/nature20150


```
          geom_boxplot(aes(group = cut_width(dim(data.sample)[1]-sampleSize, 5)),
                       outlier.size=0, outlier.shape = NA,
                   fill="white",alpha=0.5,color="black") +
          geom_abline(intercept=0,slope=0,color="black") +
          xlab(paste("Samples removed from ", dim(data.sample)[1])) + ylab("Slope")
```




```
fig2b.pl <- ggplot(selectSize.study, aes(x=dim(data.study)[1]-sampleSize, y=C.control)) +
  geom_jitter(alpha=0.3,color="grey",height=0,size=0.75) +
          scale_y_continuous(limits = c(-0.05, 0.025)) +
          geom_boxplot(aes(group = cut_width(dim(data.study)[1]-sampleSize, 1)),
                       outlier.size=0, outlier.shape = NA,
                   fill="white",alpha=0.5,color="black") +
          geom_abline(intercept=0,slope=0,color="black") +
```

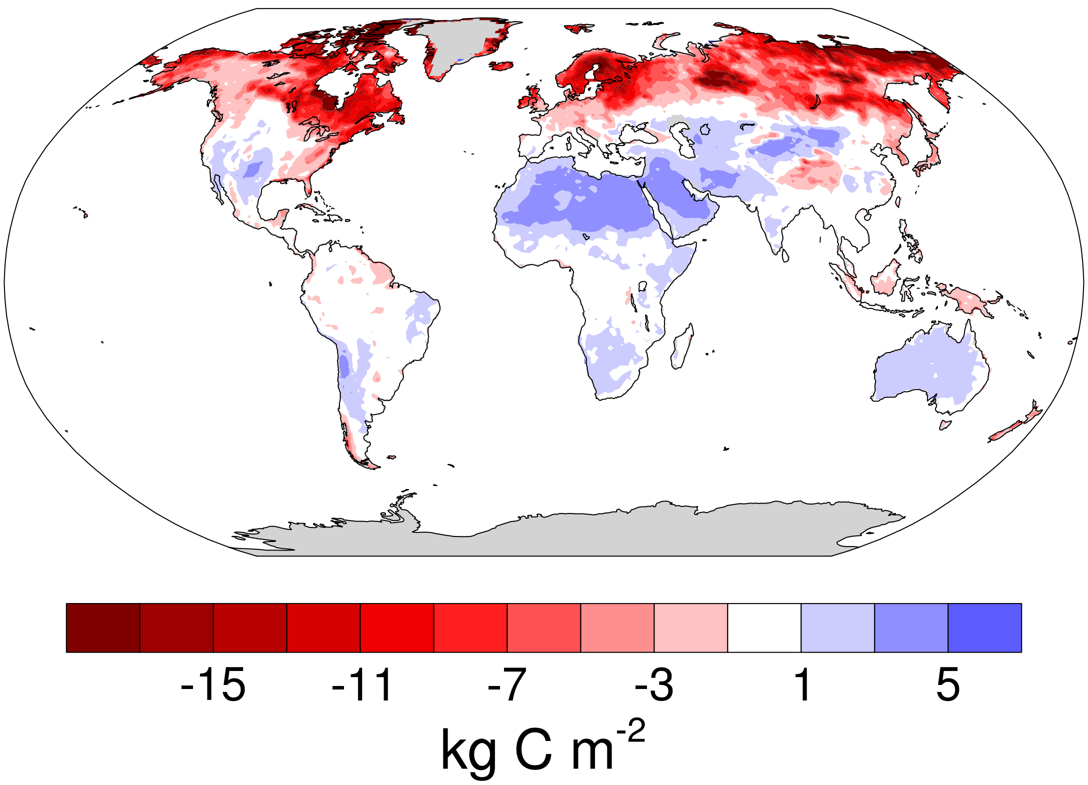
```
            xlab(paste("Studies removed from ", dim(data.study)[1])) + ylab("Slope")

fig2bTheme <- theme(axis.text.x=element_text(size=20,angle=0,colour="black"),
              axis.text.y=element_text(size=20,angle=0,colour="black"),
              axis.title=element_text(size=28),
              axis.line = element_line(colour = "black"),
              panel.grid.major = element_blank(),
              panel.grid.minor = element_blank(),
              strip.background = element_rect(colour = "black",size = 0.5),
              panel.background = element_rect(colour="black", fill="white"),
              panel.border = element_blank(),axis.ticks = element_line(colour="black"))

print(fig2b.pl + fig2bTheme)
```



```
ggsave('../figs/Figure02b.pdf', fig2b.pl + fig2bTheme, width=7, height=7, useDingbats=FALSE)
```

doi:10.1038/nature20150

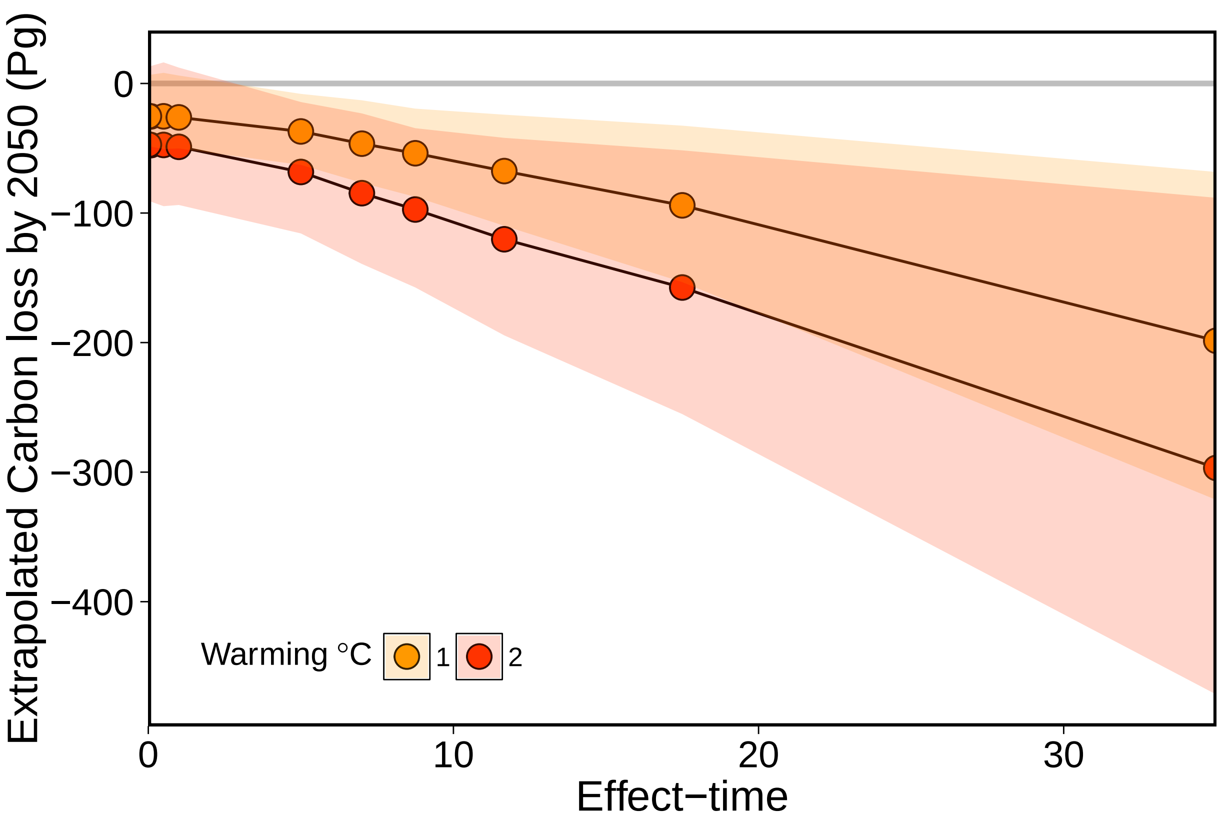
Global carbon vulnerability map (Figure 3a)



See Section "Global carbon loss map code"

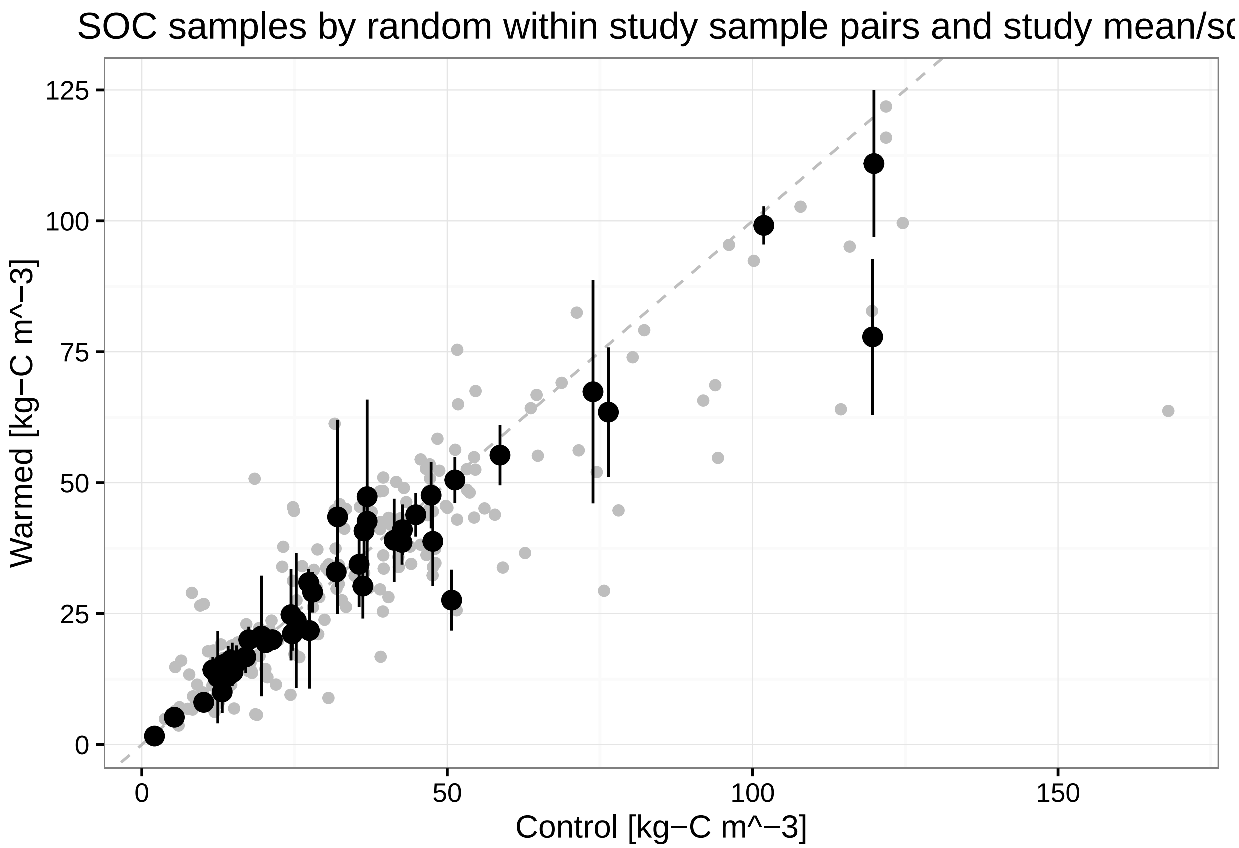
Effect-time assumptions affects soil carbon losses (Figure 3b)

```
degYrStepIntSimple.pl <- ggplot(subset(resultsTable, !grepl('dCperDeg', type) &
                                        warmingDistribution == 'CESM' &
                                        globalWarming %in% c(1,2))) +
  geom_hline(yintercept=0,col="grey",size=1) +
  geom_line(aes(x=timeStep, y=dC_qrt50, group=warming, fill=globalWarming)) +
  geom_point(aes(x=timeStep, y=dC_qrt50, fill=globalWarming), size=4, shape=21) +
  geom_ribbon(aes(x=timeStep, y=dC_qrt50, ymin=dC_qrt05, ymax=dC_qrt95,
                  fill=globalWarming, guide=NA), alpha=0.2) +
  scale_x_continuous(limits=c(0,35),expand=c(0,0))+
  scale_fill_manual(values=c('#FF9900', '#FF3300'),
                    guide = guide_legend( direction = "horizontal",
                                          title = expression("Warming"*~degree*C))) +
  labs(title='', x='Effect-time',
       y="Extrapolated Carbon loss by 2050 (Pg)") +
  theme_bw() +
  theme(axis.title=element_text(size=16),
        axis.text=element_text(size=14),
        legend.position=c(0.2,0.1),
        panel.grid.major= element_line(color=NA),
        panel.grid.minor=element_line(color=NA),
        panel.border=element_rect(color="black",fill=NA,size=1),
        axis.ticks=element_line(size=0.25),
        legend.key=element_rect(color="black",fill=NA,size=0.25))
print(degYrStepIntSimple.pl)
```



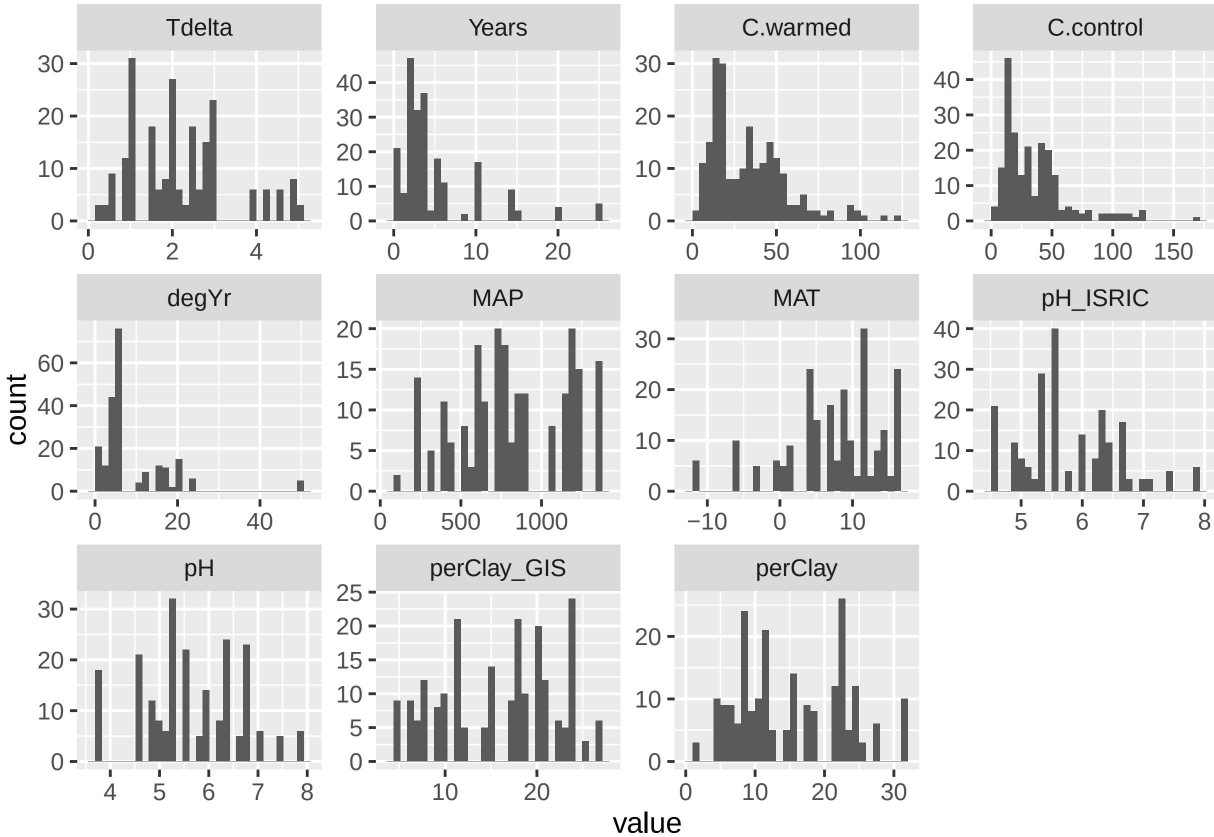
```
ggsave(degYrStepIntSimple.pl, filename='../figs/Figure03b.pdf',
       height=4.5, width=6.5, useDingbats=FALSE)
```

Data summary and basic visualizations




```
## Using Biome_GIS as id variables
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

doi:10.1038/nature20150




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

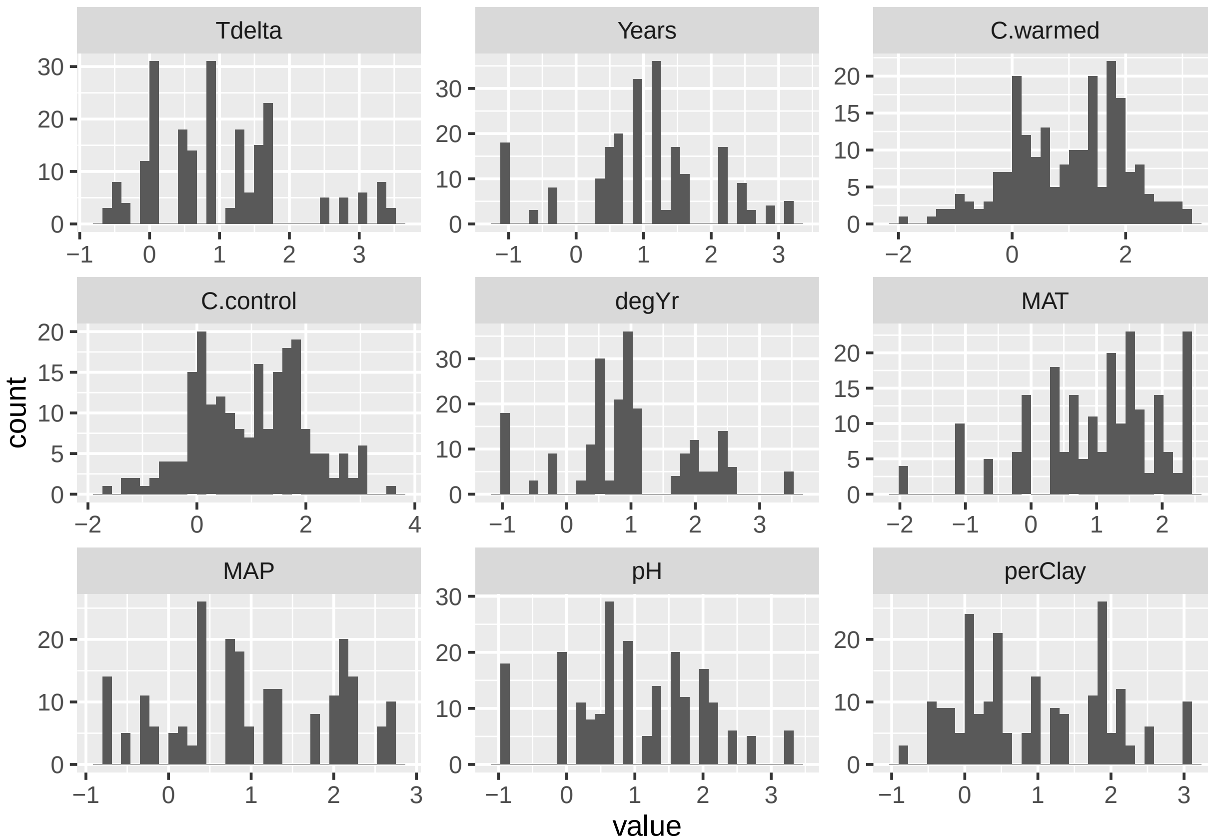


Table 6: Description of study sites including mean annual temperature (MAT), mean annual precipitation (MAP), soil pH, and soil percent clay (perClay). For standardization purposes, all climate data were collected from Bioclim and all soil data were collected from SoilGrids, unless data was provided by study.

Study Description	MAT	MAP	pH	perClay
Delta Junction, AK, USA	-3.2	298	6.6	12
Ford Forest, MI, USA	4.4	824	5.3	8
Ford Forest, MI, USA [precipitation]	4.4	824	5.3	8
FRAGILE Experiment, Svalbard, Norway [grazed]	-5.7	226	6	10
FRAGILE Experiment, Svalbard, Norway	-5.7	226	6	10
INCREASE Clocaenog, Wales, UK	7.1	1215	5.2	11
Gucheng, Hebei, China	12.7	543	7	17
Soil Warming x Nitrogen Addition Study, NH, USA	6.8	1142	4.9	7
Rocky Mountain Biological Laboratory, CO, USA	0.5	519	5.8	14
INCREASE Kiskunsag, Hungary	10.9	536	7.1	1
INCREASE Brandbjerg, Demark	8.2	603	5.5	8
Krycklan, Sweden	1	609	4.6	5
Jasper Ridge, CA, USA	13.7	635	6.2	18
Jasper Ridge, CA, USA [CO2]	13.7	635	6.2	18
Oak Ridge, Tennessee, USA	13.9	1347	5.6	27
Oak Ridge, Tennessee, USA [CO2]	13.9	1347	5.6	27

Study Description	MAT	MAP	pH	perClay
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	16.3	906	6.7	8
Oklahoma Tall Grass Prairie, OK, USA	16.3	906	6.7	8
Research Station of Songnen Grassland Ecosystem, China	5.2	436	7.9	17
Duke Forest, NC, USA [3 degrees]	14.4	1161	4.9	22
Duke Forest, NC, USA [5 degrees]	14.4	1161	4.9	22
Konza Prairie, KS, USA	12	872	6.4	24
Whitehall, GA, USA [3 degrees]	16.5	1230	4.6	21
Whitehall, GA, USA [5 degrees]	16.5	1230	4.6	21
Dry Heath Env. Control, Sweden	-0.1	390	5.1	6
Prairie Heating and CO2 Enrichment, WY, USA	7	384	7.4	23
INCREASE Garraf, Spain	15.5	632	6.8	25
HOCC-Experiment, Germany	8.9	729	6.3	22
HOCC-Experiment, Germany [precipitation 1]	8.9	729	6.3	22
HOCC-Experiment, Germany [precipitation 2]	8.9	729	6.3	22
HOCC-Experiment, Germany [precipitation 3]	8.9	729	6.8	22
HOCC-Experiment, Germany [precipitation 4]	8.9	729	6.8	22
BioCON, MN, USA [elevated C02, ambient N, negative H20]	3.8	761	3.8	11
BioCON, MN, USA [elevated C02, elevated N, negative H20]	3.8	761	3.8	11
BioCON, MN, USA [elevated C02, elevated N, ambient H20]	3.8	761	3.8	11
BioCON, MN, USA [ambient C02, ambient N, ambient H20]	3.8	761	3.8	11
BioCON, MN, USA [ambient C02, elevated N, negative H20]	3.8	761	3.8	11
BioCON, MN, USA [ambient C02, elevated N, ambient H20]	3.8	761	3.8	11
Heating of Prairie Systems 1, OR, USA	11.4	1194	5.3	31
Heating of Prairie Systems 1, OR, USA [precipitation]	11.4	1194	5.3	31
Heating of Prairie Systems 2, OR, USA [precipitation]	11.4	1364	5.5	15
Heating of Prairie Systems 3, WA, USA [precipitation]	10.1	1199	5.3	4
Heating of Prairie Systems 2, OR, USA	11.4	1364	5.5	15
Heating of Prairie Systems 3, WA, USA	10.1	1199	5.3	4
INCREASE Mols, Denmark	7.4	592	5.3	6
Arctic LTER, AK, USA	-11.2	237	6	15
Hubbard Brook, NH, USA	5.4	1082	5	9
ITEX, Greenland	-11.3	112	NA	NA
ITEX, Greenland [vegetated]	-11.3	112	NA	NA

Table 7: Mean soil carbon [kg-C m^-3] values across control study site with number of samples in each study for the control plots.

Study Description	count.control	C.control	C.sd.control
Delta Junction, AK, USA	5	32.05	NA
Ford Forest, MI, USA	3	36.19	NA
Ford Forest, MI, USA [precipitation]	3	50.72	NA
FRAGILE Experiment, Svalbard, Norway [grazed]	5	58.64	NA
FRAGILE Experiment, Svalbard, Norway	5	73.87	NA
INCREASE Clocaenog, Wales, UK	3	119.9	NA
Gucheng, Hebei, China	3	101.8	NA
Soil Warming x Nitrogen Addition Study, NH, USA	6	119.6	NA
Rocky Mountain Biological Laboratory, CO, USA	5	17.02	NA
INCREASE Kiskunsag, Hungary	3	5.32	NA
INCREASE Brandbjerg, Demark	6	10.13	NA
Krycklan, Sweden	9	44.85	NA
Jasper Ridge, CA, USA	4	14.04	NA
Jasper Ridge, CA, USA [CO2]	4	15.53	NA
Oak Ridge, Tennessee, USA	3	27.96	NA
Oak Ridge, Tennessee, USA [CO2]	3	27.32	NA
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	6	27.42	NA
Oklahoma Tall Grass Prairie, OK, USA	6	25.27	NA
Research Station of Songnen Grassland Ecosystem, China	6	20.29	NA
Duke Forest, NC, USA [3 degrees]	3	36.89	NA
Duke Forest, NC, USA [5 degrees]	3	36.89	NA
Konza Prairie, KS, USA	12	47.36	NA
Whitehall, GA, USA [3 degrees]	6	12.44	NA
Whitehall, GA, USA [5 degrees]	5	13.16	NA
Dry Heath Env. Control, Sweden	6	51.25	NA
Prairie Heating and CO2 Enrichment, WY, USA	5	17.51	NA
INCREASE Garraf, Spain	3	24.42	NA
HOCC-Experiment, Germany	4	13.84	NA
HOCC-Experiment, Germany [precipitation 1]	4	13.26	NA
HOCC-Experiment, Germany [precipitation 2]	4	11.63	NA
HOCC-Experiment, Germany [precipitation 3]	4	14.55	NA
HOCC-Experiment, Germany [precipitation 4]	4	13.95	NA
BioCON, MN, USA [elevated C02, ambient N, negative H20]	3	13.59	NA
BioCON, MN, USA [elevated C02, elevated N, negative H20]	3	19.6	NA
BioCON, MN, USA [elevated C02, elevated N, ambient H20]	3	14.91	NA

Study Description	count.control	C.control	C.sd.control
BioCON, MN, USA [ambient C02, ambient N, ambient H20]	3	14.13	NA
BioCON, MN, USA [ambient C02, elevated N, negative H20]	3	14.8	NA
BioCON, MN, USA [ambient C02, elevated N, ambient H20]	3	13.54	NA
Heating of Prairie Systems 1, OR, USA	5	42.6	NA
Heating of Prairie Systems 1, OR, USA [precipitation]	5	42.66	NA
Heating of Prairie Systems 2, OR, USA [precipitation]	5	31.82	NA
Heating of Prairie Systems 3, WA, USA [precipitation]	5	36.38	NA
Heating of Prairie Systems 2, OR, USA	5	35.57	NA
Heating of Prairie Systems 3, WA, USA	5	41.31	NA
INCREASE Mols, Denmark	3	47.64	NA
Arctic LTER, AK, USA	4	24.63	NA
Hubbard Brook, NH, USA	8	76.38	NA
ITEX, Greenland	1	2.071	NA
ITEX, Greenland [vegetated]	1	21.36	NA

Table 8: Mean soil carbon [kg-C m^-3] values across warmed study site with number of samples in each study for the warmed plots, their warming treatment [C], and length of treatment [years].

Study Description	Tdelta	Years	count.warmed	C.warmed	C.sd.warmed
Delta Junction, AK, USA	0.5	10.25	5	43.48	18.56
Ford Forest, MI, USA	4.581	5	3	30.27	6.201
Ford Forest, MI, USA [precipitation]	4.581	5	3	27.59	5.819
FRAGILE Experiment, Svalbard, Norway [grazed]	1	4	5	55.28	5.773
FRAGILE Experiment, Svalbard, Norway	1	4	5	67.37	21.31
INCREASE Clocaenog, Wales, UK	0.198	15	3	110.9	14.04
Gucheng, Hebei, China	2.34	0.6667	3	99.14	3.645
Soil Warming x Nitrogen Addition Study, NH, USA	3.989	5	5	77.85	14.91
Rocky Mountain Biological Laboratory, CO, USA	2	25	5	16.74	3.056
INCREASE Kiskunsag, Hungary	0.44	14	3	5.227	1.773
INCREASE Brandbjerg, Demark	0.9	6	6	8.075	1.399
Krycklan, Sweden	1	2	9	43.89	4.177
Jasper Ridge, CA, USA	1.773	2	4	13.09	2.205
Jasper Ridge, CA, USA [CO2]	1.773	2	4	15.65	3.289
Oak Ridge, Tennessee, USA	2.6	5	3	29.1	3.886

doi:10.1038/nature20150

Study Description	Tdelta	Years	count.warmed	C.warmed	C.sd.warmed
Oak Ridge, Tennessee, USA [CO2]	2.6	5	3	30.94	2.625
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	1.479	10	6	21.78	11.09
Oklahoma Tall Grass Prairie, OK, USA	1.479	10	6	23.69	12.93
Research Station of Songnen Grassland Ecosystem, China	1.75	3	6	19.47	0.3409
Duke Forest, NC, USA [3 degrees]	3	4	3	47.34	18.54
Duke Forest, NC, USA [5 degrees]	5	4	3	42.64	3.876
Konza Prairie, KS, USA	1	4	12	47.61	6.327
Whitehall, GA, USA [3 degrees]	2.096	3	6	12.87	8.818
Whitehall, GA, USA [5 degrees]	4.27	4	6	10.05	4.062
Dry Heath Env. Control, Sweden	1.5	14	6	50.53	4.366
Prairie Heating and CO2 Enrichment, WY, USA	2.8	6	5	20.03	2.494
INCREASE Garraf, Spain	0.94	4.5	3	24.81	8.746
HOCC-Experiment, Germany	1.954	3	4	15.47	2.457
HOCC-Experiment, Germany [precipitation 1]	1.954	3	4	15.25	1.427
HOCC-Experiment, Germany [precipitation 2]	1.954	3	4	14.28	2.466
HOCC-Experiment, Germany [precipitation 3]	1.954	3	4	16.14	2.043
HOCC-Experiment, Germany [precipitation 4]	1.954	3	4	14.81	2.861
BioCON, MN, USA [elevated C02, ambient N, negative H20]	2.5	0.42	3	13.94	2.312
BioCON, MN, USA [elevated C02, elevated N, negative H20]	2.5	0.42	3	20.74	11.53
BioCON, MN, USA [elevated C02, elevated N, ambient H20]	2.5	0.42	3	13.82	0.1774
BioCON, MN, USA [ambient C02, ambient N, ambient H20]	2.5	0.42	3	14.15	4.641
BioCON, MN, USA [ambient C02, elevated N, negative H20]	2.5	0.42	3	15.35	4.115
BioCON, MN, USA [ambient C02, elevated N, ambient H20]	2.5	0.42	3	13.9	2.389
Heating of Prairie Systems 1, OR, USA	2.75	2.2	5	38.6	4.242
Heating of Prairie Systems 1, OR, USA [precipitation]	2.75	2.2	5	41.04	4.821
Heating of Prairie Systems 2, OR, USA [precipitation]	2.98	2.16	5	32.97	2.909
Heating of Prairie Systems 3, WA, USA [precipitation]	2.94	1.75	5	40.8	6.846
Heating of Prairie Systems 2, OR, USA	2.98	2.16	5	34.42	8.201

Study Description	Tdelta	Years	count.warmed	C.warmed	C.sd.warmed
Heating of Prairie Systems 3, WA, USA	2.94	1.75	5	39.01	7.942
INCREASE Mols, Denmark	0.9	4	3	38.8	8.516
Arctic LTER, AK, USA	0.53	20	4	21.14	3.201
Hubbard Brook, NH, USA	4.83	0.8333	8	63.48	12.36
ITEX, Greenland	2	9	1	1.635	NA
ITEX, Greenland [vegetated]	2	9	1	20.02	NA

Table 9: Biome of study sites. For standardization purposes, biome allocations were generated using the UNEP biomes map, unless data was provided by study.

Study Description	Biome
Delta Junction, AK, USA	Boreal Forests/Taiga
Ford Forest, MI, USA	Temperate Broadleaf and Mixed Forests
Ford Forest, MI, USA [precipitation]	Temperate Broadleaf and Mixed Forests
FRAGILE Experiment, Svalbard, Norway [grazed]	Tundra
FRAGILE Experiment, Svalbard, Norway	Tundra
INCREASE Clocaenog, Wales, UK	Temperate Grasslands, Savannas and Shrublands
Gucheng, Hebei, China	Temperate Grasslands, Savannas and Shrublands
Soil Warming x Nitrogen Addition Study, NH, USA	Temperate Broadleaf and Mixed Forests
Rocky Mountain Biological Laboratory, CO, USA	Temperate Grasslands, Savannas and Shrublands
INCREASE Kiskunsag, Hungary	Temperate Grasslands, Savannas and Shrublands
INCREASE Brandbjerg, Demark	Temperate Broadleaf and Mixed Forests
Krycklan, Sweden	Boreal Forests/Taiga
Jasper Ridge, CA, USA	Temperate Grasslands, Savannas and Shrublands
Jasper Ridge, CA, USA [CO2]	Temperate Grasslands, Savannas and Shrublands
Oak Ridge, Tennessee, USA	Temperate Broadleaf and Mixed Forests
Oak Ridge, Tennessee, USA [CO2]	Temperate Broadleaf and Mixed Forests
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	Temperate Grasslands, Savannas and Shrublands
Oklahoma Tall Grass Prairie, OK, USA	Temperate Grasslands, Savannas and Shrublands
Research Station of Songnen Grassland Ecosystem, China	Temperate Grasslands, Savannas and Shrublands
Duke Forest, NC, USA [3 degrees]	Temperate Broadleaf and Mixed Forests
Duke Forest, NC, USA [5 degrees]	Temperate Broadleaf and Mixed Forests
Konza Prairie, KS, USA	Temperate Grasslands, Savannas and Shrublands
Whitehall, GA, USA [3 degrees]	Temperate Broadleaf and Mixed Forests
Whitehall, GA, USA [5 degrees]	Temperate Broadleaf and Mixed Forests
Dry Heath Env. Control, Sweden	Tundra

Study Description	Biome
Prairie Heating and CO2 Enrichment, WY, USA	Temperate Grasslands, Savannas and Shrublands
INCREASE Garraf, Spain	Mediterranean Forests, Woodlands and Scrub
HOCC-Experiment, Germany	Temperate Grasslands, Savannas and Shrublands
HOCC-Experiment, Germany [precipitation 1]	Temperate Grasslands, Savannas and Shrublands
HOCC-Experiment, Germany [precipitation 2]	Temperate Grasslands, Savannas and Shrublands
HOCC-Experiment, Germany [precipitation 3]	Temperate Grasslands, Savannas and Shrublands
HOCC-Experiment, Germany [precipitation 4]	Temperate Grasslands, Savannas and Shrublands
BioCON, MN, USA [elevated C02, ambient N, negative H20]	Temperate Broadleaf and Mixed Forests
BioCON, MN, USA [elevated C02, elevated N, negative H20]	Temperate Broadleaf and Mixed Forests
BioCON, MN, USA [elevated C02, elevated N, ambient H20]	Temperate Broadleaf and Mixed Forests
BioCON, MN, USA [ambient C02, ambient N, ambient H20]	Temperate Broadleaf and Mixed Forests
BioCON, MN, USA [ambient C02, elevated N, negative H20]	Temperate Broadleaf and Mixed Forests
BioCON, MN, USA [ambient C02, elevated N, ambient H20]	Temperate Broadleaf and Mixed Forests
Heating of Prairie Systems 1, OR, USA	Temperate Grasslands, Savannas and Shrublands
Heating of Prairie Systems 1, OR, USA [precipitation]	Temperate Grasslands, Savannas and Shrublands
Heating of Prairie Systems 2, OR, USA [precipitation]	Temperate Grasslands, Savannas and Shrublands
Heating of Prairie Systems 3, WA, USA [precipitation]	Temperate Grasslands, Savannas and Shrublands
Heating of Prairie Systems 2, OR, USA	Temperate Grasslands, Savannas and Shrublands
Heating of Prairie Systems 3, WA, USA	Temperate Grasslands, Savannas and Shrublands
INCREASE Mols, Denmark	Temperate Grasslands, Savannas and Shrublands
Arctic LTER, AK, USA	Tundra
Hubbard Brook, NH, USA	Temperate Broadleaf and Mixed Forests
ITEX, Greenland	Tundra
ITEX, Greenland [vegetated]	Tundra

doi:10.1038/nature20150

Helper functions

Bootstrap function

```
print(bootStrap.fn)
```

```
## function (data, myFormula, nRuns, sampleSize, lm.weights = NULL,
##     shuffleFn = NULL, numCoef, verbose = FALSE)
## {
##     sampleIndex <- matrix(NA, nrow = nRuns, ncol = sampleSize)
##     lmStats <- matrix(NA, nrow = nRuns, ncol = numCoef + 3)
##     for (ii in 1:nRuns) {
##         if (verbose)
##             cat(ii, "\n")
##         if (!is.null(shuffleFn))
##             data <- shuffleFn(data)
##         if (verbose)
##             print(head(data))
##         sampleIndex[ii, ] <- sample(1:(dim(data)[1]), size = sampleSize)
##         temp.lm <- lm(myFormula, data[sampleIndex[ii, ], ])
##         fstatArr <- summary(temp.lm)$fstatistic
##         if (verbose)
##             print(summary(temp.lm))
##         lmStats[ii, ] <- c(temp.lm$coefficients, pf(fstatArr[1],
##             fstatArr[2], fstatArr[3], lower.tail = FALSE), adj.r.squared = summary(temp.lm)$adj.r.sq
##             r.squared = summary(temp.lm)$r.squared)
##     }
##     lmStats <- as.data.frame(lmStats)
##     names(lmStats) <- c(names(temp.lm$coefficients), "p.value",
##         "adj.r.squared", "r.squard")
##     if (verbose)
##         cat("\n")
##     if (verbose)
##         print(lmStats)
##     return(lmStats)
## }
```

Read data

```
print(readSamples)
```

```
## function (filename = "../data/Soil Data Compiled_January 26.xlsx",
##     useMeanBD = TRUE, readControlMeans = FALSE)
## {
##     data <- read.xlsx2(filename, sheetIndex = 1, colIndex = c(1,
##         7, 9, 10, 11, 12))
##     names(data) <- c("Study", "Treatment", "Tdelta", "Years",
##         "perC", "bulk_density")
##     data$Tdelta <- round(data$Tdelta, 3)
```

```
##      data$perC <- round(data$perC, 3)
##      data$bulk_density <- round(data$bulk_density, 3)
##      if (useMeanBD) {
##          study.bd <- ddply(data[, c("Study", "bulk_density")],
##              .(Study), summarize, bulk_density.sd = sd(bulk_density),
##              bulk_density = mean(bulk_density))
##          data$bulk_density.sd <- NULL
##          data$bulk_density <- NULL
##          data <- merge(study.bd, data)
##      }
##      data$C <- data$perC/100 * data$bulk_density
##      data.sample <- ddply(data, c("Study", "Tdelta", "Years"),
##          function(xx) {
##              warmed <- xx$C[xx$Treatment == "W"]
##              control <- xx$C[xx$Treatment == "C"]
##              if (readControlMeans) {
##                  return(data.frame(C.warmed = warmed, C.control = mean(control)))
##              }
##              else {
##                  mismatch <- length(warmed) - length(control)
##                  if (mismatch > 0) {
##                    control <- c(control, rep(NA, mismatch))
##                  }
##                  else {
##                    warmed <- c(warmed, rep(NA, abs(mismatch)))
##                  }
##                  return(data.frame(C.warmed = warmed, C.control = sample(control)))
##              }
##          })
##      data.sample$degYr <- data.sample$Years * data.sample$Tdelta
##      return(data.sample)
## }
```

Construct study means and standard deviations

```
print(readStudyMeans)
```

```
## function (filename = "../data/Soil Data Compiled_January 26.xlsx",
##      includeBD.sd = FALSE, includeControl.sd = FALSE)
## {
##      data <- read.xlsx2(filename, sheetIndex = 1, colIndex = c(1,
##          7, 9, 10, 11, 12))
##      names(data) <- c("Study", "Treatment", "Tdelta", "Years",
##          "perC", "bulk_density")
##      data$Tdelta <- round(as.numeric(data$Tdelta), 3)
##      data$perC <- as.numeric(data$perC)
##      data$bulk_density <- as.numeric(data$bulk_density)
##      data.study <- ddply(data, .(Study, Tdelta, Years, Treatment),
##          summarize, bulk_density.sd = sd(bulk_density), bulk_density = mean(bulk_density),
##          perC.sd = sd(perC), perC = mean(perC), count = length(Treatment))
##      if (includeBD.sd) {
##          data.study$C.sd <- sqrt(data.study$perC/100^2 * data.study$bulk_density.sd^2 +
```

doi:10.1038/nature20150


```
##              data.study$perC.sd/100^2 * data.study$bulk_density^2)
##      }
##      else {
##          study.bd <- ddply(data[, c("Study", "bulk_density")],
##              .(Study), summarize, bulk_density = mean(bulk_density))
##          data.study$bulk_density.sd <- NULL
##          data.study$bulk_density <- NULL
##          data.study <- merge(study.bd, data.study)
##          data.study$C.sd <- sqrt((data.study$perC.sd/100 * data.study$bulk_density)^2)
##      }
##      data.study$C <- data.study$perC/100 * data.study$bulk_density
##      data.study <- merge(subset(data.study, Treatment == "W",
##          select = -Treatment), subset(data.study, Treatment ==
##          "C", select = -Treatment), by = c("Study", "Years", "Tdelta"),
##          suffixes = c(".warmed", ".control"))
##      if (!includeControl.sd)
##          data.study$C.sd.control <- 0
##      data.study$degYr <- data.study$Years * data.study$Tdelta
##      data.study$dC <- data.study$C.warmed - data.study$C.control
##      data.study$dC.sd <- sqrt(data.study$C.sd.warmed^2 + data.study$C.sd.control^2)
##      data.study$dC.perDegYr <- data.study$dC/data.study$degYr
##      data.study$dC.perDegYr.sd <- data.study$dC.sd/data.study$degYr
##      if (!includeControl.sd)
##          data.study$C.sd.control <- NA
##      data.study$C.se.control <- data.study$C.sd.control/data.study$count.control
##      data.study$C.se.warmed <- data.study$C.sd.warmed/data.study$count.warmed
##      data.study$dC.perDegYr.se <- data.study$dC.perDegYr.sd/sqrt(rowMeans(data.study[,
##          c("count.warmed", "count.control")]))
##      return(data.study)
## }
```

Convert R data.frame to netCDF file

```
cat(readLines('../R/Crowther_dSOC_35yr_makeNC.R'), sep = '\n')
```

```
## # Crowther_dSOC_35yr_makeNC.r
## # Will Wieder
## # July 2016
## # converts .csv to .nc file
## # data reordered go give increasing lat &  lon values
##
## library(ncdf)
## library(reshape2)
## library(raster)
## library(rgdal)
##
## #dir <- getwd() #"/Users/wwieder/Desktop/Working_files/Crowther_warming/KTB_results/"
## #setwd(dir)
## file   <- "../R/Crowther_dSOC_35yr_makeNC.R"
## fin    <- "../data/Crowther_dSOC_35yr.csv"
## Data <- read.csv(fin)
## names(Data)
```

```
##
## minLAT <- min(Data$lat)
## maxLAT <- max(Data$lat)
## minLON <- min(Data$lon)
## maxLON <- max(Data$lon)
##
## attach(Data)
## names(Data)
##
## #set up depth, lat, lon coordinates
## nLAT   <- length(as.numeric(levels(as.factor(lat))))
## nLON   <- length(as.numeric(levels(as.factor(lon))))
##
## #LAT    <- seq(minLAT,maxLAT,(90 - 89.05759))
## latDATA <- read.csv('../data/LAT.csv') # some rounding errors, read in CSV of LAT from CLM
## LAT <- latDATA$LAT
## LON    <- seq(minLON,maxLON,(360/nLON))
## nOBS   <- length(Data$dC.single)
## dims   <- c(nLAT, nLON)
##
## #something wrong w/ how lat values ordered in .csv file
## #rewrite lat so values have a regular step (as I thing they should...)
## lat2  <- rep(NA, length(lat))
## start <- 1
## for (i in 1:nLAT) {
##   end          <- start + nLON-1
##   lat2[start:end] <- LAT[i]
##   start        <- end + 1
## }
## #-------------------------------------------------------------
## #   Define Variables
## #-------------------------------------------------------------
##
## VARS  <- c('SOC','landArea','dC.single','dC.multi')
## nVARS <- length(VARS)
##                                   # close VARS loop
## gridSOC <- rasterFromXYZ(data.frame(Data$lon,lat2,Data$SOC), digits=2)
## gridSOC <- t(flip(gridSOC, direction='y') )
##
## gridArea <- rasterFromXYZ(data.frame(Data$lon,lat2,Data$landArea), digits=2)
## gridArea <- t(flip(gridArea, direction='y') )
##
## gridSingle <- rasterFromXYZ(data.frame(Data$lon,lat2,Data$dC.single), digits=2)
## gridSingle <- t(flip(gridSingle, direction='y') )
##
## gridMulti <- rasterFromXYZ(data.frame(Data$lon,lat2,Data$dC.multi), digits=2)
## gridMulti <- t(flip(gridMulti, direction='y') )
##
## #-------------------------------------------------------------
## #---------------write out .nc file----------------------------
## #-------------------------------------------------------------
## # define the netcdf coordinate variables (name, units, type)
## lat      <- dim.def.ncdf("lat","degrees_north", as.double(LAT),  create_dimvar=TRUE)
## lon      <- dim.def.ncdf("lon","degrees_east",  as.double(LON),  create_dimvar=TRUE)
```

```
## mv       <- -9999.              # missing value to use
## LATIXY   <- var.def.ncdf("LATIXY", "degrees N", list(lat), mv,
##                            longname="latitude", prec="double")
## LONGXY   <- var.def.ncdf("LONGXY", "degrees E", list(lon), mv,
##                            longname="longitude", prec="double")
## SOC_i    <- var.def.ncdf("SOC_i", units="kg C/m2", list(lon,lat), mv,
##                            longname="Soil C", prec="double")
## area     <- var.def.ncdf("Area", units="m2", list(lon,lat), mv,
##                             longname="grid_area", prec="double")
## dC_Single <- var.def.ncdf("dC_Single", units="kg C/m2", list(lon,lat), mv,
##                              longname="Single Step", prec="double")
## dC_Multi <- var.def.ncdf("dC_Multi", units="kg C/m2", list(lon,lat), mv,
##                              longname="Multi Step", prec="double")
##
## fname <- '../data/Crowther_dSOC_35y.nc'
## ncnew <- create.ncdf( fname, list(LATIXY, LONGXY, SOC_i, area, dC_Single, dC_Multi) )
##
## # Write some values to this variable on disk.
## put.var.ncdf( ncnew, LATIXY, LAT)
## put.var.ncdf( ncnew, LONGXY, LON)
## put.var.ncdf( ncnew, SOC_i,    as.array(gridSOC))
## put.var.ncdf( ncnew, area,     as.array(gridArea))
## put.var.ncdf( ncnew, dC_Single,as.array(gridSingle))
## put.var.ncdf( ncnew, dC_Multi ,as.array(gridMulti))
##
## att.put.ncdf( ncnew, 0, "created_on",date()       ,prec=NA,verbose=FALSE,definemode=FALSE )
## att.put.ncdf( ncnew, 0, "created_by","Will Wieder",prec=NA,verbose=FALSE,definemode=FALSE )
## att.put.ncdf( ncnew, 0, "created_from",fin        ,prec=NA,verbose=FALSE,definemode=FALSE )
## att.put.ncdf( ncnew, 0, "created_with",file       ,prec=NA,verbose=FALSE,definemode=FALSE )
##
## close.ncdf(ncnew)
##
## print('-------Wrote out .nc files-----------')
## print(ncnew)
```

Main analysis script

```
sessionInfo()
```

```
R version 3.2.2 (2015-08-14)
Platform: x86_64-apple-darwin13.4.0 (64-bit)
Running under: OS X 10.10.5 (Yosemite)

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] ncdf4_1.15     xlsx_0.5.7     xlsxjars_0.6.1 rJava_0.9-7
 [5] deSolve_1.12   lme4_1.1-10    Matrix_1.2-3   MASS_7.3-45
```

doi:10.1038/nature20150


```
 [9] reshape2_1.4.1 pander_0.6.0   plyr_1.8.3     ggplot2_2.0.0

loaded via a namespace (and not attached):
 [1] Rcpp_0.12.2      knitr_1.11       magrittr_1.5     splines_3.2.2
 [5] munsell_0.4.2    colorspace_1.2-6 lattice_0.20-33  minqa_1.2.4
 [9] stringr_1.0.0    tools_3.2.2      grid_3.2.2       gtable_0.1.2
[13] nlme_3.1-122     htmltools_0.2.6  yaml_2.1.13      digest_0.6.8
[17] nloptr_1.0.4     formatR_1.2.1    evaluate_0.8     rmarkdown_0.8.1
[21] labeling_0.3     stringi_1.0-1    scales_0.3.0
```

```
cat(readLines('../R/CrowtherFieldWarmingScript.R'), sep = '\n')
```

```
library(ggplot2) #make pretty plots
library(plyr) #deal with data frames nicely
library(pander) #format tables
panderOptions('table.split.table', Inf) #do not let pander split tables because bad numbering
library(reshape2) #deal with data frames nicely
library(MASS) #model selection
library(lme4) #random vs fixed effects model
library(deSolve) #solve ode
library(xlsx) #read in excel files

source('../R/bootStrap.fn.R')
source('../R/readSamples.R')
source('../R/readStudyMeans.R')

verbose <- FALSE

##Helper functions
shuffle.sample <- function(data){
  idCol <- setdiff(names(data), c('C.warmed', 'C.control'))
  return(ddply(data, idCol, summarize,
               C.warmed=sample(C.warmed, size=length(Study)),
               C.control=sample(C.control, size=length(Study))))
}

pullPvalue <- function(temp.lm){
  fstatArr <-  summary(temp.lm)$fstatistic
  return(pf(fstatArr[1], fstatArr[2], fstatArr[3], lower.tail = FALSE))
}

##Read in data
studyMeta <- read.xlsx2('../data/Soil Data Compiled_September 6, 2016.xlsx',
                        sheetIndex=2, colIndex=c(1, 9, 10, 11, 12, 13, 14, 17, 18),
                        stringsAsFactors=FALSE)
names(studyMeta) <- c('Study', 'MAP', 'MAT', 'Biome_GIS', 'Biome', 'pH_ISRIC', 'pH', 'perClay_GIS', 'pe
studyMeta$pH <- as.numeric(as.character(studyMeta$pH))
studyMeta$perClay <- as.numeric(as.character(studyMeta$perClay))

studyMeta <- studyMeta[studyMeta$Study != '',]

##Replace missing data with data products
cat('Replacing ', sum(grepl('^NA$', studyMeta$Biome)), 'of ', dim(studyMeta)[1],' missing biomes with d
studyMeta$Biome[grepl('^NA$', studyMeta$Biome)] <- studyMeta$Biome_GIS[grepl('^NA$', studyMeta$Biome)]
```

```
cat('Replacing ', sum(is.na(studyMeta$pH)), 'of ', dim(studyMeta)[1],' site missing pH values with data
studyMeta$pH[is.na(studyMeta$pH)] <- studyMeta$pH_ISRIC[is.na(studyMeta$pH)]

cat('Replacing ', sum(is.na(studyMeta$perClay)), 'of ', dim(studyMeta)[1],' site missing percent clay va
studyMeta$perClay[is.na(studyMeta$perClay)] <- studyMeta$perClay_GIS[is.na(studyMeta$perClay)]

#read study names
studyNames <- read.xlsx2('../data/Soil Data Compiled_September 6, 2016.xlsx',
                         sheetIndex=7)
names(studyNames) <- c('Study', 'Study Description')
data.sample <- readSamples(filename='../data/Soil Data Compiled_September 6, 2016.xlsx')
data.study <- readStudyMeans(filename='../data/Soil Data Compiled_September 6, 2016.xlsx')

if(!identical( setdiff(studyMeta$Study, data.sample$Study),
               setdiff(data.sample$Study, studyMeta$Study)) |
   !identical(setdiff(studyMeta$Study, studyNames$Study),
              setdiff(studyNames$Study, studyMeta$Study))){
  stop('study names do not match')
}

##Convert from g cm^-3 to kg m^-3
data.sample[, c('C.warmed', 'C.control')] <- data.sample[, c('C.warmed', 'C.control')] * 1e3
data.study[, c('bulk_density.warmed', 'C.sd.warmed', 'C.warmed', 'bulk_density.control',
               'C.sd.control',  'C.control', 'dC', 'dC.sd', 'dC.perDegYr', 'dC.perDegYr.sd',
               'C.se.control', 'C.se.warmed', 'dC.perDegYr.se')] <-
  data.study[,
             c('bulk_density.warmed', 'C.sd.warmed', 'C.warmed', 'bulk_density.control',
               'C.sd.control',  'C.control', 'dC', 'dC.sd', 'dC.perDegYr', 'dC.perDegYr.sd',
               'C.se.control', 'C.se.warmed', 'dC.perDegYr.se')] * 1e3


##Rescale data
#There is clear skew in the histograms of the years, degree-years, and carbon stocks.
#We log-transformed these variables to normalize the distribution for statistical purposes.

data.sample.plus <- merge(data.sample, studyMeta[,c('Study', 'MAT', 'MAP', 'pH', 'perClay')],
                          by='Study', all=TRUE)
data.sample.plus$degYr <- data.sample.plus$Years*data.sample.plus$Tdelta
fullRows <- apply(subset(data.sample.plus, select=-Study), c(1),
                  function(xx){all(is.finite(xx))})

if(verbose) print(sprintf('Throwing out %d samples (rows) because of missing values somewhere.',
                          sum(!fullRows)))

data.sample.plus <- data.sample.plus[fullRows,]
ggplot(melt(subset(data.sample.plus, select=-Study))) +
  geom_histogram(aes(x=value)) + facet_wrap(~variable, scale='free')
cor(subset(data.sample.plus, select=-Study))

data.sample.plus.rescaled <- data.sample.plus

data.sample.plus.rescaled$degYr <- log(data.sample.plus.rescaled$degYr)
data.sample.plus.rescaled$Years <- log(data.sample.plus$Years)
```

```
data.sample.plus.rescaled$C.control <- log(data.sample.plus$C.control)
data.sample.plus.rescaled$C.warmed <- log(data.sample.plus$C.warmed)

data.sample.plus.rescaled[,-1] <- as.data.frame(apply(
  data.sample.plus.rescaled[, -1], c(2), function(xx){
    return((xx-mean(xx, na.rm=TRUE))/sd(xx, na.rm=TRUE)+1)
  }))

##Construct LMER
lmer.list <- list(simple = lmer(C.warmed ~ C.control + (1|Study),
                    data=data.sample.plus.rescaled),
             addative.dT = lmer(C.warmed~C.control+Tdelta + (1|Study),
                    data=data.sample.plus.rescaled),
            addative.all = lmer(C.warmed~C.control+MAP+MAT+pH+degYr + perClay + (1|Study),
                    data=data.sample.plus.rescaled),
         addative.enviro = lmer(C.warmed~C.control+MAP+MAT+pH + perClay+ (1|Study),
                    data=data.sample.plus.rescaled),
          addative.treat = lmer(C.warmed~C.control+degYr + (1|Study),
                    data=data.sample.plus.rescaled),
         interactive = lmer(C.warmed~C.control*degYr+ (1|Study),
                                  data=data.sample.plus.rescaled),
         interactive.dT = lmer(C.warmed~C.control*Tdelta+ (1|Study),
                                     data=data.sample.plus.rescaled))

##Construct LM
lm.list <- list(Cw.sample = lm(C.warmed ~ C.control * degYr, data.sample),
                Cw.sample.dT = lm(C.warmed ~ C.control * Tdelta, data.sample),
                dC.sample = lm(C.warmed - C.control ~ C.control * degYr, data.sample),
                dC.dT.sample = lm(C.warmed - C.control ~ C.control * Tdelta, data.sample),
                dCperDegYr.sample = lm((C.warmed-C.control)/(Years*Tdelta) ~ C.control,
                                           data.sample),
                dCperDeg.sample = lm((C.warmed-C.control)/Tdelta ~ C.control,
                                         data.sample),
                Cw.study = lm(C.warmed ~ C.control * degYr, data.study),
                Cw.study.dT = lm(C.warmed ~ C.control * Tdelta, data.study),
                dC.study = lm(C.warmed - C.control ~ C.control * degYr, data.study),
                dC.dT.study = lm(C.warmed - C.control ~ C.control * Tdelta, data.study),
                dCperDegYr.study = lm((C.warmed-C.control)/(Years*Tdelta) ~ C.control,
                                          data.study),
                dCperDeg.study = lm((C.warmed-C.control)/Tdelta ~ C.control,
                                        data.study))

modelFits <- ldply(lm.list,
                   function(xx){
                     data.frame(model=as.character(xx$call)[2],
                                data=as.character(xx$call)[3],
                                adjR2 = sprintf('%0.3f', summary(xx)$adj.r.squared),
                                pvalue=sprintf('%0.3g', pullPvalue(xx)))
                   })


##Sample model vs data distributions
interactive.model <- function(pars=summary(lm.list$Cw.study)$coefficients,
                              C.control, C.sd.control, degYr){
```

```
  C_degYr.par <- rnorm(1, mean=pars['C.control:degYr', 'Estimate'],
                       sd=pars['C.control:degYr', 'Std. Error'])
  C.par <- rnorm(1, mean=pars['C.control', 'Estimate'], sd=pars['C.control', 'Std. Error'])
  degYr.par <- rnorm(1, mean=pars['degYr', 'Estimate'], sd=pars['degYr', 'Std. Error'])
  inter.par <- rnorm(1, mean=pars['(Intercept)', 'Estimate'],
                     sd=pars['(Intercept)', 'Std. Error'])
  model <- inter.par+ C.par*C.control + degYr.par*degYr + C_degYr.par*C.control*degYr

  return(model)
}

modelData.df <- data.frame()
for(ii in 1:1000){
  modelData.df <- rbind(modelData.df,
                        data.frame(index = 1:length(data.study$C.warmed),
                                   rnd.data=rnorm(n=length(data.study$C.warmed),
                                                  mean=data.study$C.warmed,
                                                  sd=data.study$C.sd.warmed),
                                   rnd.model =
                                     interactive.model(C.control=data.study$C.control,
                                                       C.sd.control=data.study$C.sd.control,
                                                       degYr=data.study$degYr)))
}

summaryMD.df <- ddply(modelData.df, 'index', summarize,
                      data.mean=mean(rnd.data), data.sd=sd(rnd.data),
                      model.mean=mean(rnd.model), model.sd=sd(rnd.model))

##bootstrap slope
selectSize.sample <- adply(floor(seq(10, dim(data.sample)[1], length=50)), c(1),
                           function(xx){
                             ans <- bootStrap.fn(
                                myFormula=(C.warmed-C.control)/(Years*Tdelta) ~ C.control,
                                data=data.sample, nRuns=100, sampleSize=xx, numCoef=2,
                                shuffleFn=shuffle.sample)
                             ans$sampleSize <- xx
                             return(ans)
                           })


selectSize.study <- adply(3:(dim(data.study)[1]), c(1),
                          function(xx){
                            ans <- bootStrap.fn(
                               myFormula=(C.warmed-C.control)/(Years*Tdelta) ~ C.control,
                               data=data.study, nRuns=100, sampleSize=xx, numCoef=2)
                            ans$sampleSize <- xx
                            return(ans)
                          })

##Pull CI for parameters from subset samples
dCperDeg.boot <- bootStrap.fn(
  myFormula=(C.warmed-C.control)/Tdelta ~ C.control,
  data=data.sample, nRuns=1e3, sampleSize=200, numCoef=2, shuffleFn=shuffle.sample)
```

```
dCperDegYr.boot <- bootStrap.fn(
  myFormula=(C.warmed-C.control)/(Years*Tdelta) ~ C.control,
  data=data.sample, nRuns=1e3, sampleSize=200, numCoef=2, shuffleFn=shuffle.sample)


dCperDegYr.mod.boot <- llply(list(wk1=1/52, mon1 = 1/12, mon6 = 6/12, yr1 = 1,
                                  yr4 = 4, yr5 = 5,  yr7 = 7, yr8 = 8,
                                  yr8.75= 8.75, yr10 = 10, yr11.6=35/3,
                                  yr15 = 15,
                                  yr17.5=17.5, yr20 = 20, yr25 = 25, yr30 = 30, yr35 = 35),
                         function(xx){
                           data.sample$Years.mod <- data.sample$Years
                           data.sample$Years.mod[data.sample$Years.mod > xx] <- xx
                           ans <-  bootStrap.fn(
                          myFormula = (C.warmed-C.control)/(Years.mod*Tdelta) ~ C.control,
                             data=data.sample, nRuns=1e3, sampleSize=200,
                             numCoef=2, shuffleFn=shuffle.sample, verbose=FALSE)
                           return(ans)
                         })


parKDE <- kde2d(dCperDegYr.boot$C.control, dCperDegYr.boot$`(Intercept)`, n=100)
parBins <- melt(parKDE$z)
parBins <- subset(parBins, value > max(value)*0.01)
parBins$slope <- parKDE$x[parBins$Var1]
parBins$intercept <- parKDE$y[parBins$Var2]
parBins$alpha <- parBins$value/max(parBins$value)

parRange <- ldply(c(list(dCperDegYr = dCperDegYr.boot,
                       dCperDeg = dCperDeg.boot),
                       dCperDegYr.mod.boot), function(xx){
                       ans <- as.data.frame(apply(xx, c(2),
                                quantile, c(0.05, 0.5, 0.95)))
                       ans$qrt <- c(0.05, 0.5, 0.95)
                       return(ans)
                       })

names(parRange)[1:3] <- c('type','intercept', 'C')
save(file='../data/parCIforLM.RData', parRange)
```

Extrapolation code

```
cat(readLines('../R/globalExtrapolations.R'), sep='\n')
```

```
###Set up
library(ncdf4)
library(ggplot2)
library(plyr)
verbose <- FALSE
dataDir <- '../data/'
readIn.tsl <- TRUE
```

```
#############################
###Read in maps
inputs.ls <- list(soilGrid=list(filename='SoilGrids_0.9x1.25.nc',
                                varName='OCSTHA_M',
                                units='tonnes ha^-1', #convertion factor 1/10 for kg m^-2
                                depthWeight=c(1, 1, 0, 0, 0, 0)),
                  #mid points c(2.5  10.0  22.5  45.0  80.0 150.0) cm
                  #implies 5cm, 10cm, 15cm, 30cm, 60cm, 60cm layer lengths
                  #take top 15cm

                  HWSD=list(filename='surfdata_0.9x1.25_simyr2000_c120906_HWSD_soil.nc',
                            varName='DOM_SOC', #dominatent mapping unit;
                            #alt area weighted AWT_SOC
                            units='kg C m^-2',
                            depthWeight=c(1, 0)), #0-30 cm, 30-70 cm soil layers
                  landfrac=list(filename='sftlf_fx_CESM1-BGC_historical_r0i0p0.nc',
                                varName='sftlf',
                                units='percent'),
                  gridArea=list(filename='areacella_fx_CESM1-BGC_historical_r0i0p0.nc',
                                varName='areacella',
                                units='m2'))

maps.ls <- lapply(inputs.ls, function(args){
  ncin <- nc_open(sprintf('%s%s', dataDir, args$filename))
  if(verbose) print(ncin)
  lon <- ncvar_get(ncin, 'lon') #longitude
  lat <- ncvar_get(ncin, 'lat') #longitude
  ans <- ncvar_get(ncin, args$varName)
  nc_close(ncin)

  if(!is.null(args$depthWeight)){
    ans <- apply(ans, c(1,2), function(xx){sum(args$depthWeight*xx)})
  }

  dimnames(ans) <- list(lon=lon, lat=lat)
  ans <- as.data.frame.table(ans, stringsAsFactors=FALSE, responseName='value')
  ans <- as.data.frame(lapply(ans, as.numeric))

  return(ans)
})

maps.ls$landArea <- merge(maps.ls$gridArea, maps.ls$landfrac,
                          by=c('lon', 'lat'), suffixes=c('.area', '.perc'))
maps.ls$landArea$value <- maps.ls$landArea$value.area*maps.ls$landArea$value.perc/100

if(readIn.tsl){
  #CESM1-BGC Soil Temperature
  ##Pre-processing in cdo
  ##$cdo yearmean tsl_Lmon_CESM1-BGC_rcp85_r1i1p1_200601-204912.nc
  ##              tsl_yrmean_CESM1-BGC_rcp85_r1i1p1_200601-204912.nc
  ##$cdo sellevidx,1,2,3,4 tsl_yrmean_CESM1-BGC_rcp85_r1i1p1_200601-204912.nc temp.nc
  ##$cdo vertmean temp.nc tsl_yrShortMean_CESM1-BGC_rcp85_r1i1p1_200601-204912.nc

  ncin <- nc_open(sprintf('%stsl_yrShortMean_CESM1-BGC_rcp85_r1i1p1_200601-204912.nc',
```

```
                          dataDir))
  if(verbose) print(ncin)
  tsl <- ncvar_get(ncin, 'tsl') #units K
  lon <- ncvar_get(ncin, 'lon') #longitude
  lat <- ncvar_get(ncin, 'lat') #longitude
  time <- ncvar_get(ncin, 'time') #days since 2005-1-1 0:0:0
  nc_close(ncin)

  dimnames(tsl) <- list(lon=lon, lat=lat, yr=(time/365) + 2005)
  tsl <- as.data.frame.table(tsl, stringsAsFactors=FALSE, responseName='value')
  tsl <- as.data.frame(lapply(tsl, as.numeric))

  ##Make the latitudes aggree, off by 1e-6
  tsl$lat <- round(tsl$lat, 2)
  maps.ls <- lapply(maps.ls, function(xx){xx$lat <- round(xx$lat, 2); return(xx)})


  ##Trim tsl to only cover 2015-2049
  tsl <- subset(tsl, yr >= 2015 & yr <=2049)
  tsl.start <- ddply(subset(tsl, yr >= min(yr) & yr < (min(yr)+10)), .(lon, lat),
                     summarize, value=mean(value))
  tsl.end <-  ddply(subset(tsl, yr > max(yr)-10 & yr <= max(yr)), .(lon, lat),
                    summarize, value=mean(value))
  tsl.change <- merge(tsl.start, tsl.end, by=c('lon', 'lat'), suffixes=c('.inital', '.final'))

  if(verbose){
    print(ggplot(tsl.change) + geom_raster(aes(x=lon, y=lat, fill=value.final-value.inital)) +
            labs(title='CESM-BCG temperature change'))
    print(ggplot(tsl.change) + geom_histogram(aes(x=value.final-value.inital)) +
            labs(title='CESM-BCG temperature change'))
  }
}

if(verbose){
  print(ggplot(maps.ls$soilGrid) + geom_raster(aes(x=lon, y=lat, fill=value/10)) +
          scale_fill_continuous(limits=c(0, 300),low="yellow", high='red') +
          labs( title='Soil Grids'))
  print(ggplot(maps.ls$HWSD) + geom_raster(aes(x=lon, y=lat, fill=value)) +
          scale_fill_continuous(limits=c(0, 100),low="yellow", high='red') + labs(title='HWSD'))

  print(ggplot(maps.ls$landfrac) + geom_raster(aes(x=lon, y=lat, fill=value/100)) +
          scale_fill_continuous(limits=c(0, 1),low="yellow", high='red') +
          labs( title='Land Fraction'))
  print(ggplot(maps.ls$gridArea) + geom_raster(aes(x=lon, y=lat, fill=value)) +
          labs( title='Grid Area'))
  print(ggplot(maps.ls$landArea) + geom_raster(aes(x=lon, y=lat, fill=value)) +
          labs( title='Land Area'))

}

##########################################
##Make one dataframe to work from so that the lat-lon pair up appropreately
##########################################
commonGrid <- merge(maps.ls$landArea,
```

```
                    merge(maps.ls$soilGrid, maps.ls$HWSD,
                          by=c('lon', 'lat'), suffixes=c('.SG', '.H')),
                    by=c('lon', 'lat'))
if(readIn.tsl){
  commonGrid <- merge(tsl.change, commonGrid,
                      by=c('lon', 'lat'), suffixes=c('.Dtsl', '.landArea'))
}

commonGrid <- rename(commonGrid, c('value.inital'='inital.temperature',
                                   'value.final'='final.temperature',
                                   'value.area'='cell.area',
                                   'value.perc'='land.percentage',
                                   'value'='land.area',
                                   'value.SG'='SoilGrid.SOC', 'value.H'='HWSD.SOC'))

##Shift the units for soil grid to kg m^-2
commonGrid$SoilGrid.SOC <- commonGrid$SoilGrid.SOC/10

###Remove 0 values
##commonGrid$SoilGrid.SOC[commonGrid$SoilGrid.SOC == 0] <- NA
##commonGrid$HWSD.SOC[commonGrid$HWSD.SOC == 0] <- NA

commonGrid$allFinite <- is.finite(rowSums(subset(commonGrid, select=-HWSD.SOC))) &
  commonGrid$land.area != 0

#################################
###Pull temperature normalization from CESM if needed
#################################
if(readIn.tsl){
  globalCESM.dT <- with(commonGrid, sum(land.area*
                                        (final.temperature-inital.temperature)*allFinite,
                                      na.rm=TRUE)/sum(land.area*allFinite, na.rm=TRUE))
}else{
  globalCESM.dT <- NA
}

if(verbose){
  ggplot(commonGrid) + geom_raster(aes(x=lon, y=lat, fill=allFinite)) +
    labs(title='Shared grid cells')
  print(sprintf("Global totals: HWSD = %0.2f Pg,
                SoilGrid = %0.2f Pg, inital T = %0.2f C, dT = %0.2f C",
                with(commonGrid, sum(land.area*(HWSD.SOC)*allFinite, na.rm=TRUE)/1e12),
                with(commonGrid, sum(land.area*(SoilGrid.SOC)*allFinite, na.rm=TRUE)/1e12),
                ifelse(readIn.tsl, with(commonGrid,
                                        sum(land.area*inital.temperature*allFinite, na.rm=TRUE)/
                                          sum(land.area*allFinite, na.rm=TRUE))-273.15, NA),
                globalCESM.dT
  ))
}

##############################
###Run the global extrapolation
###############################
load(sprintf('%sparCIforLM.RData', dataDir))
```

```
soilDepth <- 0.15 #in m; for HWSD it's 0.3
##Number of years we run through
runTime <- 35

dC <- function(args, step, Cstock){
  #correct for soil depth but converting stocks from per area to per volume
  #...and then correcting the result from per volume to per area
  return(step*(args$C*Cstock/soilDepth+args$intercept)*soilDepth)
}

##Use the temperature change distribution from CESM from year 2040-2049 and 2015-2024
if(readIn.tsl){
  degWarmedRate.ls <- list(oneDeg=1/runTime, twoDeg=2/runTime,
                           threeDeg=3/runTime, fourDeg=4/runTime,
                     oneDeg_CESM_normed = (commonGrid$final.temperature-
                                           commonGrid$inital.temperature)/
                       globalCESM.dT*1/runTime,
                     twoDeg_CESM_normed = (commonGrid$final.temperature-
                                           commonGrid$inital.temperature)/
                       globalCESM.dT*2/runTime,
                     threeDeg_CESM_normed = (commonGrid$final.temperature-
                                             commonGrid$inital.temperature)/
                       globalCESM.dT*3/runTime,
                     fourDeg_CESM_normed = (commonGrid$final.temperature-
                                            commonGrid$inital.temperature)/
                       globalCESM.dT*4/runTime)
}else{
  degWarmedRate.ls <-list(oneDeg=1/runTime, twoDeg=2/runTime)
}

#Time step for each linear model type
dtime.ls <- list(wk1=1/52, mon1 = 1/12, mon6 = 6/12, yr1 = 1,
                 yr4 = 4, yr5 = 5,  yr7 = 7, yr8 = 8,
                 yr8.75= 8.75, yr10 = 10, yr11.6=35/3,
                 yr17.5=17.5, yr20 = 20, yr25 = 25, yr30 = 30, yr35 = 35)

resultsFull <- ldply(degWarmedRate.ls, .id='warming', function(degWarmedRate){
  ##Calculate the SOC losses
  SOC.losses <- ddply(parRange, c('type', 'qrt', 'intercept', 'C'),
    function(xx){
      #cat(xx$type)

      C.map <- commonGrid$SoilGrid.SOC

      if(grepl('^dCperDegYr$', xx$type)){
        dC.map <- ldply(dtime.ls, .id=NULL, function(warmedTime){
          degStep <- degWarmedRate/2*warmedTime^2
          return(data.frame(degYr.mean=sum(degStep*commonGrid$land.area, na.rm=TRUE)/
                              sum(is.finite(degStep)*commonGrid$land.area, na.rm=TRUE),
                            timeStep=warmedTime,
                            lon=commonGrid$lon,
                            lat=commonGrid$lat,
                            value.C=C.map,
```

doi:10.1038/nature20150


```
                              landArea=commonGrid$land.area*commonGrid$allFinite,
                              value.dC=dC(args=xx, step=degStep, Cstock=C.map)))
        })
      }else if(grepl('^dCperDeg$', xx$type)){
        dC.map <- data.frame(degYr.mean=NA,
                             timeStep=NA,
                             lon=commonGrid$lon,
                             lat=commonGrid$lat,
                             value.C=C.map,
                             landArea=commonGrid$land.area*commonGrid$allFinite,
                             value.dC=dC(args=xx, step=degWarmedRate*runTime, Cstock=C.map))
      }else{ ##Cap study
        #print(xx$type)
        #print(!(xx$type %in% names(dtime.ls)) || (runTime/dtime.ls[[xx$type]]) %% 1 != 0)
        if(!(xx$type %in% names(dtime.ls)) ||
              (runTime/dtime.ls[[xx$type]]) %% 1 != 0){
          return(data.frame()) #don't run if you can't cover the entire period
        }
        runningC <- C.map
        degStep <- degWarmedRate/2*dtime.ls[[xx$type]]^2 #cumulative degYr for each time step
        for(ii in seq(0, runTime-1, by=dtime.ls[[xx$type]])){
          runningC <- runningC + dC(args=xx, step=degStep, Cstock=runningC)
        }

        dC.map <- data.frame(degYr.mean=mean(degStep, na.rm=TRUE),
                             timeStep=dtime.ls[[xx$type]],
                             lon=commonGrid$lon,
                             lat=commonGrid$lat,
                             value.C=C.map,
                             landArea=commonGrid$land.area*commonGrid$allFinite,
                             value.dC=runningC-C.map)
      }

      ##max loss is the inital carbon stock
      dC.map$value.dC[is.finite(dC.map$value.C+dC.map$value.dC) & dC.map$value.dC +
                        dC.map$value.C < 0] <-
        -1*dC.map$value.C[is.finite(dC.map$value.C+dC.map$value.dC) & dC.map$value.dC +
                            dC.map$value.C < 0]

      #dC.map <- merge(dC.map, commonGrid[,c('lon', 'lat', 'land.area', 'allFinite')])
      return(ddply(dC.map, c('timeStep', 'degYr.mean'),
                   summarize, dC=sum(value.dC*landArea, na.rm=TRUE)/1e12))
    }) #end SOC.losses
  }) #end resultsTable

resultsTable <- merge(subset(resultsFull, qrt==0.95,
                             select=c('warming', 'type', 'timeStep', 'degYr.mean', 'dC')),
  merge(subset(resultsFull, qrt==0.05,
               select=c('warming', 'type', 'timeStep','degYr.mean',  'dC')),
              subset(resultsFull, qrt==0.50,
                     select=c('warming', 'type', 'timeStep', 'degYr.mean', 'dC')),
              by=c('warming', 'type', 'degYr.mean', 'timeStep'), suffixes=c('_qrt05', '_qrt50')))
resultsTable <- rename(resultsTable, c('dC'='dC_qrt95'))
```

```
resultsTable$dodge.timeStep <- resultsTable$timeStep +
  rnorm(n=length(resultsTable$timeStep), mean=0, sd=0.1)
deg.key <- list("fourDeg"=4,  "oneDeg"=1,   "threeDeg"=3, "twoDeg"=2)
resultsTable$globalWarming <- as.factor(unlist(lapply(strsplit(
  as.character(resultsTable$warming), split="_"), function(xx){deg.key[[xx[[1]]]]})))

resultsTable$warmingDistribution <- unlist(lapply(strsplit(
  as.character(resultsTable$warming), split="_"),
  function(xx){ifelse(length(xx) > 1, 'CESM', 'unif')}))

save(file='../data/globalExtrapolations.RData', resultsTable, resultsFull)

#####################
##Make plots
degYrSingle.pl <- ggplot(subset(resultsTable, grepl('dCperDegYr', type))) +
  geom_point(aes(x=timeStep, y=dC_qrt50)) +
  geom_errorbar(aes(x=timeStep, y=dC_qrt50, ymin=dC_qrt05, ymax=dC_qrt95)) +
  facet_wrap(~warming, nrow=2) +
  labs(title='dC per degree-year across single time steps', x='years', y='Pg C')
ggsave(degYrSingle.pl, filename='../figs/degYrSingleTimeStep.pdf')

degYr.pl <- ggplot(subset(resultsTable, grepl('dCperDegYr', type))) +
  geom_point(aes(x=degYr.mean, y=dC_qrt50, color=grepl('CESM', warming))) +
  geom_ribbon(aes(x=degYr.mean, y=dC_qrt50, ymin=dC_qrt05, ymax=dC_qrt95,
                  fill=grepl('CESM', warming)), alpha=0.3) +
  scale_fill_discrete(guide=guide_legend(title='CESM'))+guides(color=FALSE) +
  labs(title='dC per degree-year across single time steps', x='degree-years', y='Pg C')
ggsave(degYr.pl, filename='../figs/degYr.pdf')

degSingle.pl <- ggplot(subset(resultsTable, 'dCperDeg'== type)) +
  geom_point(aes(x=warming, y=dC_qrt50)) +
  geom_errorbar(aes(x=warming, y=dC_qrt50, ymin=dC_qrt05, ymax=dC_qrt95)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) + labs(title='dC per degree')
ggsave(degSingle.pl, filename='../figs/degSingleTimeStep.pdf')


degYrStepInt.pl <- ggplot(subset(resultsTable, !grepl('dCperDeg', type))) +
  geom_line(aes(x=timeStep, y=dC_qrt50, group=warming, linetype=warmingDistribution)) +
  geom_ribbon(aes(x=timeStep, y=dC_qrt50, ymin=dC_qrt05, ymax=dC_qrt95, group=warming),
              alpha=0.2) +
  facet_wrap(~globalWarming)
ggsave(degYrStepInt.pl, filename='../figs/degYrMultiTimeStep.pdf')

##See Crowther2016Sup.Rmd for figure code
write.csv(file='../data/degYrMultiTimeStepSimple.csv',
          subset(resultsTable, !grepl('dCperDeg', type) & globalWarming %in% c('1', '2')))

singleStep.pl <- ggplot(subset(resultsTable, grepl('dCperDeg', type) &
                                 globalWarming %in% c(1,2) &
                                 (is.na(timeStep) | timeStep == 35))) +
  geom_point(aes(x=globalWarming, y=dC_qrt50, color=type, shape=warmingDistribution), cex=5) +
  geom_errorbar(aes(x=globalWarming, y=dC_qrt50, color=type, linetype=warmingDistribution,
                    ymin=dC_qrt05, ymax=dC_qrt95)) +
  labs(title='Soil carbon losses at 35 years, one step', x='Average temperature increase',
```

```
       y='Global change in soil carbon [Pg C]')
ggsave(singleStep.pl, filename='../figs/singleStep.pdf')
write.csv(file='../data/singleStep.csv',
          subset(resultsTable, grepl('dCperDeg', type) &
                   globalWarming %in% c('1', '2') &
                   (is.na(timeStep) | timeStep == 35), -dodge.timeStep))

########
##Make ncdf file for pretty maps
Cshift <- data.frame(lon=commonGrid$lon, lat=commonGrid$lat,
                     SOC=commonGrid$SoilGrid.SOC,
                     landArea=commonGrid$land.area, #*commonGrid$allFinite,
                     dC.single=dC(args=subset(parRange, type=='dCperDegYr' & qrt==0.5),
                                  step=degWarmedRate.ls$oneDeg_CESM_normed/2*35^2,
                                  Cstock=commonGrid$SoilGrid.SOC))

runningC <- Cshift$SOC
degStep <- degWarmedRate.ls$oneDeg_CESM_normed/2*1^2 #cumulative degYr for each time step
for(ii in seq(0, runTime-1, by=1)){
  runningC <- runningC + dC(args=subset(parRange, type=='yr1' & qrt==0.5),
                            step=degStep, Cstock=runningC)
}
Cshift$dC.multi <- runningC-Cshift$SOC

negFlag <- is.finite(Cshift[, 'dC.single'] + Cshift[, 'SOC']) &
  (Cshift[, 'dC.single'] + Cshift[, 'SOC'] < 0 )
Cshift[negFlag, 'dC.single'] <- -1*Cshift[negFlag, 'SOC']
negFlag <- is.finite(Cshift[, 'dC.multi'] + Cshift[, 'SOC']) &
  (Cshift[, 'dC.multi'] + Cshift[, 'SOC'] < 0 )
Cshift[negFlag, 'dC.multi'] <- -1*Cshift[negFlag, 'SOC']

cat('Single step: ', sum(Cshift$dC.single*Cshift$landArea, na.rm=TRUE)/1e12, '=?=',
    unlist(subset(resultsTable, grepl('dCperDegYr', type) &
                    globalWarming %in% c(1, 2) &
                    (is.na(timeStep) | timeStep == 35) &
                    warming=='oneDeg_CESM_normed', dC_qrt50)),
    '\nOne yr step: ', sum(Cshift$dC.multi*Cshift$landArea, na.rm=TRUE)/1e12, '=?=',
    unlist(subset(resultsTable, !grepl('dCperDeg', type) & globalWarming %in% c(1, 2) &
                    type=='yr1' & warming=='oneDeg_CESM_normed', dC_qrt50)), '\n')

write.csv(file='Crowther_dSOC_35yr.csv', Cshift)
```

Global carbon loss map code

```
cat(readLines('../ncl/plot_warming_loss.ncl'), sep='\n')
```

```
; July 2016
; Will Wieder
; plots changes in SOC stocks from Kathe's analyses.
; *********************************************************************

load "$NCARG_LIB/ncarg/nclscripts/csm/gsn_code.ncl"
```

```
load "$NCARG_LIB/ncarg/nclscripts/csm/gsn_csm.ncl"
load "$NCARG_LIB/ncarg/nclscripts/csm/contributed.ncl"
load "$NCARG_LIB/ncarg/nclscripts/csm/shea_util.ncl"

begin
;---------------------------------------------------------------
;Read in input variables
;--------------------------------------------------------------
 path       = (/"/project/tss/wwieder/soilCN/global_run/warming/"/)
 fin        = path + "Crowther_dSOC_35y.nc"

 data       = addfile(fin, "r")
 SGRD_SOC   = data->SOC_i(:,:)                ; SoilGrids SOC pools, kgC/m2, 0-15 cm
 area       = data->Area
 dC_single  = data->dC_Single
 dC_multi   = data->dC_Multi

 glob_SOCi  = sum(SGRD_SOC  * area) / 1.e12
 glob_dC_s  = sum(dC_single * area) / 1.e12
 glob_dC_m  = sum(dC_multi  * area) / 1.e12

 print(glob_SOCi)
 print(glob_dC_s)
 print(glob_dC_m)

end

;************************************************
; plot SOC losses
; Fig. 3 in manuscript
;************************************************
 fout = path + "Crowther_dSOC_35y_step_wZERO"
 wks  = gsn_open_wks("ps"  , fout); open a X11 or ps file

 res                          = True
 res@gsnDraw                  = False
 res@gsnFrame                 = False
 res@cnSmoothingOn            = False
 res@mpProjection             = "Robinson"
 res@mpOutlineOn              = True
 res@lbOrientation            = "Horizontal"
 res@mpPerimOn                = False
 res@mpGridAndLimbOn          = True
 res@mpGridLatSpacingF        = 180
 res@mpGridLonSpacingF        = 180
 res@mpGridLineThicknessF     = 0.
 res@mpGridLineColor          = "transparent"
 res@mpGridMaskMode           = "MaskLand"

  gsn_define_colormap(wks,"BlWhRe")
  res@gsnSpreadColors     = True                   ; use full colormap
  res@gsnSpreadColorEnd   =  68                ; start with last color
;  res@gsnSpreadColorStart =  2                 ; start with last color
   gsn_reverse_colormap(wks)                        ; reverse colormap
```

```
  res@gsnLeftString        = ""
  res@gsnRightString       = ""
  res@cnFillOn             = True
  res@cnLinesOn            = False         ; Turn lines off
  res@cnLineLabelsOn       = False         ; Turn labels off
  res@cnLevelSelectionMode = "ManualLevels"
  res@cnMinLevelValF       = -17  ;  -3.75*5
  res@cnMaxLevelValF       =  5.  ; 0.50*5
  res@cnLevelSpacingF      =  2.  ; 0.5*5
  res@lbLabelStrings       = (/-17.,-15.,-13.,-11.,-9.,-7,-5.,-3.,-1,1.,3.,5./)
;  res@lbLabelStrings       = (/-17.,-13.,-9.,-5.,-1.,1.,5/)
  res@lbLabelFontHeightF   =  0.025                ; make labels larger
  res@lbTitleOn            = True                  ; turn on title
  res@lbTitlePosition      = "Bottom"
  res@lbTitleString        = "kg C m~S~-2~N "
  res@lbTitleFontHeightF   = .030                  ; make title smaller
  res@pmLabelBarOrthogonalPosF = .05               ; move whole thing down

  res@vpXF             = 0.1                 ; make plot bigger
  res@vpYF             = 0.9
  res@vpWidthF         = 0.8
  res@vpHeightF        = 0.8
  plot                 = gsn_csm_contour_map(wks,dC_single,res)
  resP                   = True                    ; modify the panel plot
  resP@gsnFrame          = False                   ; don't advance panel plot
  gsn_panel(wks,plot,(/1,1/),resP)                 ; now draw as one plot
  frame(wks)


  print("wrote "+fout+".ps")
  delete([/plot, res, resP, wks,fout/])
```