Supplement of Biogeosciences, 15, 3659–3671, 2018
https://doi.org/10.5194/bg-15-3659-2018-supplement

Biogeosciences Open Access EGU

Supplement of

Field-warmed soil carbon changes imply high 21st-century modeling uncertainty

Katherine Todd-Brown et al.

Correspondence to: Katherine Todd-Brown (katherine.todd-brown@pnnl.gov)

Contents

Analytical proofs

Below we review three analytical analysis of traditional soil decomposition models given 1) independent pool structure (no carbon transfer between the pools), 2) cascade pool structure (carbon moving from pools with fast turnover times to pools with slow turnover times), and 3) full feedback models (carbon moves both between fast to slow and slow to fast turnover pools). Three claims are examined: 1. Given soil decomposition model with independent pools the change in carbon stock for a given time period is dominated by a single pool. 2. For cascade (carbon moving strictly from 'fast' to 'slow' carbon pools) and independent (no carbon exchange between pools) decomposition models there exists a unique indepent pool model. 3. There exists

a unique one pool decay rate (total outputs divided by total stocks) for any multipool soil decomposition model at steady state.

Change in carbon stock of independent pools

Claim: Given soil decomposition model with independent pools the change in carbon stock for a given time period is dominated by a single pool.

First consider a single carbon pool model:

$$\frac{dC_i}{dt} = u_{in} - k_i C_i$$

where C_i is the soil carbon stock, t is the time, u_{in} the soil carbon inputs, and k_i the decay rate such that $k_i = 1/\tau_i$ where τ_i is the turnover time.

This leads to the solution

$$C_i(t) = \frac{u_{in}}{k_i} - a_i e^{-k_i t}$$

such that $a_i = \frac{u_{in}}{k_i} - C_i(t=0)$ or the difference between metastable state (equivlant to steady state when inputs are constant) and the intial condition. If the turnover time of each pool is an order of magnitude different such that $k_1 >> k_2 >> .. >> k_z$. For a given time t_m we can split the pools into three catagories, those with 'fast' decay rates $k_n >> \frac{1}{t_m}$, those with 'slow' decay rates $k_j << \frac{1}{t_m}$, and those on the same time scale $k_m \approx \frac{1}{t_m}$. For fast pools, $e^{-k_n t_m} \approx 0$ implying that $C_n(t_m) = \frac{u_{in}}{k_n}$. For slow pools, $e^{-k_j t_m} \approx 1$ implying that $C_j(t_m) = \frac{u_{in}}{k_j} - a_j$. Thus the sum of all the pools can be writen as follows:

$$\sum C_i(t_m) = \sum_{n<m} \frac{u_{in}}{k_n} + (\frac{u_{in}}{k_m} - a_m e^{-k_m t_m}) + \sum_{m<j} (\frac{u_{in}}{k_j} - a_j)$$

$$\ldots = \sum_{n<m} \frac{u_{in}}{k_n} + (\frac{u_{in}}{k_m} - a_m e^{-k_m t_m}) + \sum_{m<j} C_j(t=0)$$

Thus the change in total soil carbon from $t=0$ to $t=t_m$ can be written as:

$$\sum C_i(t_m) - \sum C_i(t=0) = \sum_{n<m} (\frac{u_{in}}{k_n} - C_n(t=0)) + (\frac{u_{in}}{k_m} - a_m e^{-k_m t_m} - C_m(t=0)) + \sum_{m<j} (C_j(t=0) - C_j(t=0))$$

$$\Delta C = \sum_{n \leq m} a_n - a_m e^{-k_m t_m}$$

In other words, the change in total soil carbon is the sum of the shift in metastable state of the fast pools plus a portion of the metastabel shift in the pool of the timescale of interest, which is a one pool model.

Change in carbon stock for cascade pools

Claim: For cascade (carbon moving strictly from 'fast' to 'slow' carbon pools) and independent (no carbon exchange between pools) decomposition models there exists a unique indepent pool model.

Note that if a matrix is triangular and the diagonal entries are distinct, then it is diagonalizable. This condition hold for soil decomposition models which are cascade or independent pool structures.

Consider the linear ODE system:

$$\frac{\mathrm{d}\boldsymbol{C}(t)}{\mathrm{d}t} = u_{in}\boldsymbol{b} - \boldsymbol{K}\boldsymbol{A}\boldsymbol{C}(t)$$

with initial condition $\boldsymbol{C}(t=0) = \boldsymbol{C}_0$. For simplicty, denote $\boldsymbol{B} = \boldsymbol{K}\boldsymbol{A}$. In the special case when $\boldsymbol{B}$ is triangular and has distinct diagonal entries, we know that $\boldsymbol{B}$ is diagonalizable, i.e.,

$$\boldsymbol{B} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^{-1},$$

where

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix},$$

and $0 < \lambda_1 < \lambda_2 < \cdots < \lambda_n$. Let $\tilde{\boldsymbol{C}} = \boldsymbol{P}^{-1}\boldsymbol{C}$, we have

$$\frac{\mathrm{d}\tilde{\boldsymbol{C}}(t)}{\mathrm{d}t} = u_{in}\boldsymbol{P}^{-1}\boldsymbol{b} - \boldsymbol{\Lambda}\tilde{\boldsymbol{C}}(t),$$

which means that we have n decoupled one-pool models and can be solved independently, i.e.,

$$\frac{\mathrm{d}\tilde{\boldsymbol{C}}_i(t)}{\mathrm{d}t} = u_{in}(\boldsymbol{P}^{-1}\boldsymbol{b})_i - \lambda_i\tilde{\boldsymbol{C}}_i(t),\ i = 1, \ldots, n.$$

Therefor, for any cascade decomposition model there exists a unique indepedent pool model whose pools are a linear combination of the orginal described by $\boldsymbol{P}^{-1}\boldsymbol{C}$.

We can thus apply the analysis in the first section to arrive at the conclusion that states that at a particular time scale of interest this multi-pool model can be well described using a single pool.

One and multi pool decay rates at steady state

Claim: There exists a unique one pool decay rate (total outputs divided by total stocks) for any multipool soil decomposition model at steady state.

Consider the single pool model:

$$\frac{\mathrm{d}C(t)}{\mathrm{d}t} = u_{in} - kC(t),$$

where the change in soid carbon stock (C) over time (t) is equal to the soil carbon inputs (u_{in}) minus the soil carbon stock times the bulk decomposition rate (k). At steady-state,

$$C = \frac{u_{in}}{k}.$$

For an analogous multipool model described by the following linear ODE system:

$$\frac{\mathrm{d}\boldsymbol{C}(t)}{\mathrm{d}t} = u_{in}\boldsymbol{b} - \boldsymbol{K}\boldsymbol{A}\boldsymbol{C}(t)$$

Note, the matrix $\boldsymbol{B} := \boldsymbol{K}\boldsymbol{A}$ is an M-matrix, i.e., its entries $\{b_{i,j}\}_{i,j=1,\ldots,n}$ satisfy $b_{ii} > 0$ and $b_{ij} \leq 0$ for $i \neq j$.

The steady-state solution for an M-matrix is given by,

$$\boldsymbol{C} = (\boldsymbol{K}\boldsymbol{A})^{-1}(u_{in}\boldsymbol{b}).$$

Let $\tilde{C} := \sum_i \boldsymbol{C}_i$, then

$$\tilde{C} = \sum_{j=1}^{n} \left(\sum_{i=1}^{n} [(\boldsymbol{K}\boldsymbol{A})^{-1}]_{i,j} \right) (u_{in}\boldsymbol{b}_j).$$

Note that because $\boldsymbol{K}\boldsymbol{A}$ is an M-matrix, its inverse is a non-negative matrix, i.e., all entries of $(\boldsymbol{K}\boldsymbol{A})^{-1}$ are non-negative. Hence,

$$\sum_{j=1}^{n} \left(\sum_{i=1}^{n} [(\boldsymbol{K}\boldsymbol{A})^{-1}]_{i,j} \right) \boldsymbol{b}_j > 0.$$

So, we can write

$$\tilde{C} = \frac{u_{in}}{\left(\sum_{j=1}^{n}\left(\sum_{i=1}^{n}[(\boldsymbol{KA})^{-1}]_{i,j}\right)\boldsymbol{b}_j\right)^{-1}}$$

Thus the analogous decay rate of the one pool approximation of the linear set of ODE's in a soil decomposition model is

$$k = \left(\sum_{j=1}^{n}\left(\sum_{i=1}^{n}[(\boldsymbol{KA})^{-1}]_{i,j}\right)\boldsymbol{b}_j\right)^{-1}$$

A word on temperature sensitivity

There are several soil temperature sensitivty functions that are in use in the CMIP5 Earth system model. In general, these functions are well approximated by the Q_{10} function described below for most environmental temperature ranges. Soil temperature senstivities typically are measured as a multiplicative increase (Q_{10}) in decay rate (k) for 10 degrees C of warming from reference or

$$k_w = k_c Q_{10}^{(T-T_{ref})/10}$$

where k_w is the warmed decay rate k_c is the control decay rate, T the warmed temperature, T_{ref} the reference temperature, and Q_{10} the factor of increase.

If we assume that each soil carbon pool has a unique Q_{10} and allow the temperature T to vary with time, we can now construct a diagonal matrix $\boldsymbol{Q(t)}$ who's main diagonals are of the form $q_i^{(T(t)-T_{ref})/10}$ allowing us to add a temperature sensitivity to the previous decomposition equation

$$\frac{\mathrm{d}\boldsymbol{C}(t)}{\mathrm{d}t} = u_{in}\boldsymbol{b} - \boldsymbol{KQ(t)AC}(t)$$

If we now derive a new bulk decay rate with this temeprature dependency we see that the bulk decay rate is no longer temporally constant:

$$k = \left(\sum_{j=1}^{n}\left(\sum_{i=1}^{n}[(\boldsymbol{Q(t)KA})^{-1}]_{i,j}\right)\boldsymbol{b}_j\right)^{-1} = \left(\sum_{j=1}^{n}\left(\sum_{i=1}^{n}[(\boldsymbol{KA})^{-1}]_{i,j}\right)\boldsymbol{q_{jj}}^{-(T(t)-T_{ref})/10}\boldsymbol{b}_j\right)^{-1}$$

Thus the bulk decay rate k does not show a Q_{10} multiplicative relationship with temperature in this case.

While there has been some support for different Q_{10} values for soils of different recalcentrency (ie a unique Q_{10} for each soil carbon pool), the CMIP5 Earth system models apply a single temperature sensitivity to each pool. For a summary of the Earth system model temperature sensitivity functions, please see the main paper.

Because this function is a uniform multiplier across all pools, when we consider the one pool approximation we can factor this out without loss of generality. If the Q_{10} factor was unique for each soil pool then the bulk Q_{10} would be a weighted sum of the indiviual pools, complicating the analysis.

Field Q10 calculations

```
library(tidyverse)
library(pander)
library(ncdf4) #read in netCDF files
```

```
library(mapproj) #map things nicely
library(reshape2)
library(gridExtra)
library(cowplot)
sessionInfo()
```

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Sierra 10.12.6
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
## 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] cowplot_0.9.2   gridExtra_2.3   reshape2_1.4.3  mapproj_1.2.6
##  [5] maps_3.3.0      ncdf4_1.16      pander_0.6.1    forcats_0.3.0
##  [9] stringr_1.3.1   dplyr_0.7.4     purrr_0.2.4     readr_1.1.1
## [13] tidyr_0.8.0     tibble_1.4.2    ggplot2_2.2.1   tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
##  [1] haven_1.1.1      lattice_0.20-35  colorspace_1.3-2 htmltools_0.3.6
##  [5] yaml_2.1.19      rlang_0.2.0      pillar_1.2.2     foreign_0.8-70
##  [9] glue_1.2.0       modelr_0.1.2     readxl_1.1.0     bindrcpp_0.2.2
## [13] bindr_0.1.1      plyr_1.8.4       munsell_0.4.3    gtable_0.2.0
## [17] cellranger_1.1.0 rvest_0.3.2      psych_1.8.4      evaluate_0.10.1
## [21] knitr_1.20       parallel_3.5.0   broom_0.4.4      Rcpp_0.12.16
## [25] scales_0.5.0     backports_1.1.2  jsonlite_1.5     mnormt_1.5-5
## [29] hms_0.4.2        digest_0.6.15    stringi_1.2.2    grid_3.5.0
## [33] rprojroot_1.3-2  cli_1.0.0        tools_3.5.0      magrittr_1.5
## [37] lazyeval_0.2.1   crayon_1.3.4     pkgconfig_2.0.1  xml2_1.2.0
## [41] lubridate_1.7.4  assertthat_0.2.0 rmarkdown_1.9    httr_1.3.1
## [45] rstudioapi_0.7   R6_2.2.2         nlme_3.1-137     compiler_3.5.0
```

Read in data files

```
studyMeta <- readxl::read_xlsx('data/Soil Data Compiled_September 6, 2016.xlsx',
                               sheet=2) %>%
  select(c(1, 9, 10, 11, 12, 13, 14, 17, 18)) %>%
  rename(MAP = `Precipitation (avg. annual mm)`,
         MAT = `Temperature (avg. annual degrees C)`,
         Biome_GIS = Biome, Biome = `Author Updated Biome`,
         pH_ISRIC = `pH (from ISRIC)`,
         pH = `Author Updated pH`,
         perClay_GIS = `% clay`,
         perClay = `Author Updated % clay`) %>%
```

```
  mutate_at(c('pH_ISRIC', 'pH', 'perClay_GIS', 'perClay'), as.numeric)

##Replace missing data with data products
cat('Replacing ', sum(grepl('^NA$', studyMeta$Biome)), 'of ', dim(studyMeta)[1],
    ' missing biomes with data product.\n',
    'Replacing ', sum(is.na(studyMeta$pH)), 'of ', dim(studyMeta)[1],
    ' site missing pH values with data product.\n',
    'Replacing ', sum(is.na(studyMeta$perClay)), 'of ', dim(studyMeta)[1],
    ' site missing percent clay values with data product.\n')
```

```
## Replacing  31 of  49  missing biomes with data product.
##  Replacing  41 of  49  site missing pH values with data product.
##  Replacing  35 of  49  site missing percent clay values with data product.
```

```
studyMeta <- studyMeta %>%
  mutate(Biome = if_else(grepl('^NA$', Biome), Biome_GIS, Biome),
         pH = if_else(is.na(pH), pH_ISRIC, pH),
         perClay=if_else(is.na(perClay), perClay_GIS, perClay))

#read study names
studyNames <- readxl::read_xlsx('data/Soil Data Compiled_September 6, 2016.xlsx',
                          sheet=7) %>%
  rename(Study = `Old Name`, Study.name = `New Name`)

data.study <- readxl::read_xlsx('data/Soil Data Compiled_September 6, 2016.xlsx',
                                sheet = 1) %>%
  select(Study, `Treatment (W=warming, C=control)`, `Mean delta T (degrees C)`,
         `Length of Study (years)`, `%C`, `bulk density (g/cm^3)`) %>%
  rename(Treatment=`Treatment (W=warming, C=control)`,
         Tdelta=`Mean delta T (degrees C)`,
         Years=`Length of Study (years)`,
         perC=`%C`,
         bulk_density=`bulk density (g/cm^3)`) %>%
  mutate(Tdelta=round(Tdelta, 3)) %>% #there are some precision errors in the spreadsheet
  gather(measure, value, perC, bulk_density) %>%
  group_by(Study, Tdelta, Years, Treatment, measure) %>%
  summarize(mean=mean(value), sd=sd(value), count = length(value))

cat('Removing ', length(unique(data.study$Study[data.study$Years <= 2])), 'of',
    length(unique(data.study$Study)),
    'studies that are shorter then 2 yrs.\n')
```

```
## Removing  13 of 49 studies that are shorter then 2 yrs.
```

```
data.study <- filter(data.study, Years > 2)

cat('There are', length(unique(data.study$Study)), 'sites in this study.')
```

```
## There are 36 sites in this study.
```

```
data.study.long <- data.study

data.study <- full_join((data.study.long %>%
                           filter(measure == 'perC') %>%
                           select(Study, Tdelta, Years, Treatment, mean) %>%
                           spread(Treatment, mean) %>%
```

```
                          rename(perC.control=C, perC.warmed=W)),
                        (data.study.long %>%
                          filter(measure == 'perC') %>%
                          select(Study, Tdelta, Years, Treatment, sd) %>%
                          spread(Treatment, sd) %>%
                          rename(perC.control.sd=C, perC.warmed.sd=W)),
                        by=c('Study', 'Tdelta', 'Years'))

studyMeta %>% filter(Study %in% data.study$Study) %>% group_by(Biome) %>% tally %>% print
```

```
## # A tibble: 5 x 2
##   Biome                                            n
##   <chr>                                        <int>
## 1 Boreal Forests/Taiga                             1
## 2 Mediterranean Forests, Woodlands and Scrub       1
## 3 Temperate Broadleaf and Mixed Forests           10
## 4 Temperate Grasslands, Savannas and Shrublands   18
## 5 Tundra                                           6
```

For a complete description of the data with mixed-model analysis see Crowther et al, Nature 2016.

Table S1: Field data summary

```
myTableS1 <- data.study %>%
  left_join(studyNames, by='Study') %>% ungroup() %>%
    select(Study.name, Tdelta, Years, perC.control,
           perC.control.sd, perC.warmed, perC.warmed.sd) %>%
    rename(`Study Name`=Study.name, `Treatment temperature (C)`=Tdelta,
           `Length of treatement (yr)` = Years,
           `Percent control organic carbon`= perC.control,
           `Control sd` =  perC.control.sd,
           `Percent warmed organic carbon`= perC.warmed,
           `Warmed sd` =  perC.warmed.sd)

myTableS1 <- as.data.frame(myTableS1)
row.names(myTableS1) <- myTableS1$`Study Name`
myTableS1$`Study Name` <- NULL
```

```
pander(myTableS1, digits=2)
```

Table 1: Table continues below

	Treatment temperature (C)
Delta Junction, AK, USA	0.5
Ford Forest, MI, USA	4.6
Ford Forest, MI, USA [precipitation]	4.6
FRAGILE Experiment, Svalbard, Norway [grazed]	1
FRAGILE Experiment, Svalbard, Norway	1
INCREASE Clocaenog, Wales, UK	0.2
Soil Warming x Nitrogen Addition Study, NH, USA	4

	Treatment temperature (C)
Rocky Mountain Biological Laboratory, CO, USA	2
INCREASE Kiskunsag, Hungary	0.44
INCREASE Brandbjerg, Demark	0.9
Oak Ridge, Tennessee, USA	2.6
Oak Ridge, Tennessee, USA [CO2]	2.6
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	1.5
Oklahoma Tall Grass Prairie, OK, USA	1.5
Research Station of Songnen Grassland Ecosystem, China	1.8
Duke Forest, NC, USA [3 degrees]	3
Duke Forest, NC, USA [5 degrees]	5
Konza Prairie, KS, USA	1
Whitehall, GA, USA [3 degrees]	2.1
Whitehall, GA, USA [5 degrees]	4.3
Dry Heath Env. Control, Sweden	1.5
Prairie Heating and CO2 Enrichment, WY, USA	2.8
INCREASE Garraf, Spain	0.94
HOCC-Experiment, Germany	2
HOCC-Experiment, Germany [precipitation 1]	2
HOCC-Experiment, Germany [precipitation 2]	2
HOCC-Experiment, Germany [precipitation 3]	2
HOCC-Experiment, Germany [precipitation 4]	2
Heating of Prairie Systems 1, OR, USA	2.8
Heating of Prairie Systems 1, OR, USA [precipitation]	2.8
Heating of Prairie Systems 2, OR, USA [precipitation]	3
Heating of Prairie Systems 2, OR, USA	3
INCREASE Mols, Denmark	0.9
Arctic LTER, AK, USA	0.53
ITEX, Greenland	2
ITEX, Greenland [vegetated]	2

Table 2: Table continues below

	Length of treatement (yr)
Delta Junction, AK, USA	10
Ford Forest, MI, USA	5
Ford Forest, MI, USA [precipitation]	5
FRAGILE Experiment, Svalbard, Norway [grazed]	4
FRAGILE Experiment, Svalbard, Norway	4
INCREASE Clocaenog, Wales, UK	15

	Length of treatement (yr)
Soil Warming x Nitrogen Addition Study, NH, USA	5
Rocky Mountain Biological Laboratory, CO, USA	25
INCREASE Kiskunsag, Hungary	14
INCREASE Brandbjerg, Demark	6
Oak Ridge, Tennessee, USA	5
Oak Ridge, Tennessee, USA [CO2]	5
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	10
Oklahoma Tall Grass Prairie, OK, USA	10
Research Station of Songnen Grassland Ecosystem, China	3
Duke Forest, NC, USA [3 degrees]	4
Duke Forest, NC, USA [5 degrees]	4
Konza Prairie, KS, USA	4
Whitehall, GA, USA [3 degrees]	3
Whitehall, GA, USA [5 degrees]	4
Dry Heath Env. Control, Sweden	14
Prairie Heating and CO2 Enrichment, WY, USA	6
INCREASE Garraf, Spain	4.5
HOCC-Experiment, Germany	3
HOCC-Experiment, Germany [precipitation 1]	3
HOCC-Experiment, Germany [precipitation 2]	3
HOCC-Experiment, Germany [precipitation 3]	3
HOCC-Experiment, Germany [precipitation 4]	3
Heating of Prairie Systems 1, OR, USA	2.2
Heating of Prairie Systems 1, OR, USA [precipitation]	2.2
Heating of Prairie Systems 2, OR, USA [precipitation]	2.2
Heating of Prairie Systems 2, OR, USA	2.2
INCREASE Mols, Denmark	4
Arctic LTER, AK, USA	20
ITEX, Greenland	9
ITEX, Greenland [vegetated]	9

Table 3: Table continues below

	Percent control organic carbon	Control sd
Delta Junction, AK, USA	10	4.7
Ford Forest, MI, USA	4.6	0.81
Ford Forest, MI, USA [precipitation]	6.7	3.1
FRAGILE Experiment, Svalbard, Norway [grazed]	26	4.9

	Percent control organic carbon	Control sd
FRAGILE Experiment, Svalbard, Norway	22	6.6
INCREASE Clocaenog, Wales, UK	40	1.2
Soil Warming x Nitrogen Addition Study, NH, USA	18	4.1
Rocky Mountain Biological Laboratory, CO, USA	4.9	0.94
INCREASE Kiskunsag, Hungary	0.38	0.095
INCREASE Brandbjerg, Demark	1.1	0.21
Oak Ridge, Tennessee, USA	1.9	0.18
Oak Ridge, Tennessee, USA [CO2]	1.8	0.15
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	2	1.5
Oklahoma Tall Grass Prairie, OK, USA	2	1.2
Research Station of Songnen Grassland Ecosystem, China	1.6	0.062
Duke Forest, NC, USA [3 degrees]	3.4	0.7
Duke Forest, NC, USA [5 degrees]	3.4	0.7
Konza Prairie, KS, USA	4.7	0.67
Whitehall, GA, USA [3 degrees]	1.2	0.54
Whitehall, GA, USA [5 degrees]	1.3	0.59
Dry Heath Env. Control, Sweden	40	2.6
Prairie Heating and CO2 Enrichment, WY, USA	1.7	0.14
INCREASE Garraf, Spain	2.1	0.24
HOCC-Experiment, Germany	1	0.063
HOCC-Experiment, Germany [precipitation 1]	1.1	0.08
HOCC-Experiment, Germany [precipitation 2]	0.93	0.23
HOCC-Experiment, Germany [precipitation 3]	1.1	0.15
HOCC-Experiment, Germany [precipitation 4]	1.1	0.092
Heating of Prairie Systems 1, OR, USA	7.1	0.47
Heating of Prairie Systems 1, OR, USA [precipitation]	7.1	1
Heating of Prairie Systems 2, OR, USA [precipitation]	3.1	0.28
Heating of Prairie Systems 2, OR, USA	3.4	0.32
INCREASE Mols, Denmark	4.8	0.56
Arctic LTER, AK, USA	45	3
ITEX, Greenland	0.19	NA
ITEX, Greenland [vegetated]	1.6	NA

	Percent warmed organic carbon	Warmed sd
Delta Junction, AK, USA	14	6
Ford Forest, MI, USA	3.8	0.78

	Percent warmed organic carbon	Warmed sd
Ford Forest, MI, USA [precipitation]	3.6	0.76
FRAGILE Experiment, Svalbard, Norway [grazed]	24	2.5
FRAGILE Experiment, Svalbard, Norway	20	6.5
INCREASE Clocaenog, Wales, UK	37	4.7
Soil Warming x Nitrogen Addition Study, NH, USA	11	2.2
Rocky Mountain Biological Laboratory, CO, USA	4.8	0.87
INCREASE Kiskunsag, Hungary	0.37	0.13
INCREASE Brandbjerg, Demark	0.87	0.15
Oak Ridge, Tennessee, USA	2	0.26
Oak Ridge, Tennessee, USA [CO2]	2.1	0.18
Oklahoma Tall Grass Prairie, OK, USA [clipped grass]	1.6	0.81
Oklahoma Tall Grass Prairie, OK, USA	1.8	1
Research Station of Songnen Grassland Ecosystem, China	1.5	0.027
Duke Forest, NC, USA [3 degrees]	4.3	1.7
Duke Forest, NC, USA [5 degrees]	3.9	0.35
Konza Prairie, KS, USA	4.8	0.63
Whitehall, GA, USA [3 degrees]	1.3	0.88
Whitehall, GA, USA [5 degrees]	0.99	0.4
Dry Heath Env. Control, Sweden	40	3.4
Prairie Heating and CO2 Enrichment, WY, USA	1.9	0.24
INCREASE Garraf, Spain	2.1	0.75
HOCC-Experiment, Germany	1.1	0.18
HOCC-Experiment, Germany [precipitation 1]	1.2	0.11
HOCC-Experiment, Germany [precipitation 2]	1.1	0.2
HOCC-Experiment, Germany [precipitation 3]	1.2	0.15
HOCC-Experiment, Germany [precipitation 4]	1.1	0.22
Heating of Prairie Systems 1, OR, USA	6.4	0.71
Heating of Prairie Systems 1, OR, USA [precipitation]	6.8	0.8
Heating of Prairie Systems 2, OR, USA [precipitation]	3.2	0.28
Heating of Prairie Systems 2, OR, USA	3.3	0.79
INCREASE Mols, Denmark	3.9	0.85
Arctic LTER, AK, USA	38	5.8
ITEX, Greenland	0.15	NA
ITEX, Greenland [vegetated]	1.5	NA

Table S1: Percent organic carbon data across studies with treatment information.

Define model score for a given Q10

```
##This returns the model-data metrics for a given parameter
fitSSModel <- function(pars, Cw=data.study$perC.warmed,
                       Cc=data.study$perC.control,
                       dT=data.study$Tdelta){
  Q10 <- pars$Q10
  temp.lm <- summary(lm(data ~ model, data.frame(data=Cw, model=Cc*Q10^(-dT/10))))

  ans <- data.frame(
    norm_RMSE=sqrt( sum( ((Cw-(Cc*Q10^(-dT/10)))/Cw)^2 )/length(Cw) ),
    RMSE=sqrt( sum( (Cw-(Cc*Q10^(-dT/10)))^2 )/length(Cw) ),
    R2=temp.lm$adj.r.squared,
    intercept=temp.lm$coefficients['(Intercept)','Estimate'],
    intercept.sd=temp.lm$coefficients['(Intercept)','Std. Error'],
    slope=temp.lm$coefficients['model','Estimate'],
    slope.sd=temp.lm$coefficients['model','Std. Error'])
  return(ans)
}
```

Subset data, randomize, and score Q10s

```
removeSitesFitSSModel <- function(param){

  index <- sample.int(nrow(data.study), size=param$selectSize)

  #calculate mode fit
  ans <- fitSSModel(pars=param,
                    Cw=data.study$perC.warmed[index],
                    dT=data.study$Tdelta[index],
                    Cc=data.study$perC.control[index])
  ans$type <- 'True'

  #randomly assign case-control labels
  useTrueCase <- rbinom(length(index), size=1, prob=0.5)
  Cw_set <- ifelse(useTrueCase==1, data.study$perC.warmed[index],
                   data.study$perC.control[index])
  Cc_set <- ifelse(useTrueCase==1, data.study$perC.control[index],
                   data.study$perC.warmed[index])

  rand_CC <- fitSSModel(pars=param,
                        Cw=Cw_set, dT=data.study$Tdelta[index], Cc=Cc_set)
  rand_CC$type <- 'Random C-C'
  ans <- rbind.fill(ans, rand_CC)

  # #randomly assign temperature treatments
  # rand_dT <- fitSSModel(pars=param,
  #                       Cw=data.study$perC.warmed[index],
  #                       dT=sample(data.study$Tdelta[index]),
  #                       Cc=data.study$perC.control[index])
  # rand_dT$type <- 'Random dT'
  # ans <- rbind.fill(ans, rand_dT)
```

```
  return(ans)
}

newSamples <- 1000
startIndex <- 0
if(file.exists('temp/removeSites.RData')){
  load('temp/removeSites.RData')
  old_removeSites <- removeSites
  newSamples <- newSamples - length(unique(subset(old_removeSites,
                                                  selectSize == 5 & Q10 == 0.1)$index))
  startIndex <- max(old_removeSites$index)

  Q10seq <- unique(old_removeSites$Q10)
  selectSizeSeq <- unique(old_removeSites$selectSize)
}else{
  Q10seq <- seq(0.1, 5, by=0.1)
  selectSizeSeq <- c(seq(5, nrow(data.study)-10, by=5), nrow(data.study) - c(8, 6, 4:2))
}

if(newSamples > 0){
  removeSites <- expand.grid(list(Q10=Q10seq, selectSize=rep(selectSizeSeq, newSamples)))
  removeSites$index <- startIndex + (1:nrow(removeSites))

  removeSites <- ddply(removeSites, c('Q10', 'selectSize', 'index'),
                       removeSitesFitSSModel)

  if(file.exists('temp/removeSites.RData')){
    removeSites <- rbind(old_removeSites, removeSites)
    rm(old_removeSites)
  }

  save(removeSites, file='temp/removeSites.RData')
}else{
  rm(old_removeSites)
}

removeSites <- removeSites %>%
  filter(type != 'Random dT')
```

Scoring model-data fits

```
trueDist <- removeSites %>%
  filter(type=='True', selectSize==34) %>%
  select(-type, -selectSize, -norm_RMSE) %>%
  group_by(Q10) %>%
  sample_n(size=50) %>%
  mutate(intercept.min = intercept - 2*intercept.sd,
         slope.min = slope - 2*slope.sd,
         intercept.max = intercept + 2*intercept.sd,
         slope.max = slope + 2*slope.sd) %>%
  gather(longMetric, value, contains('RMSE'), R2,
         starts_with('intercept'), starts_with('slope')) %>%
```

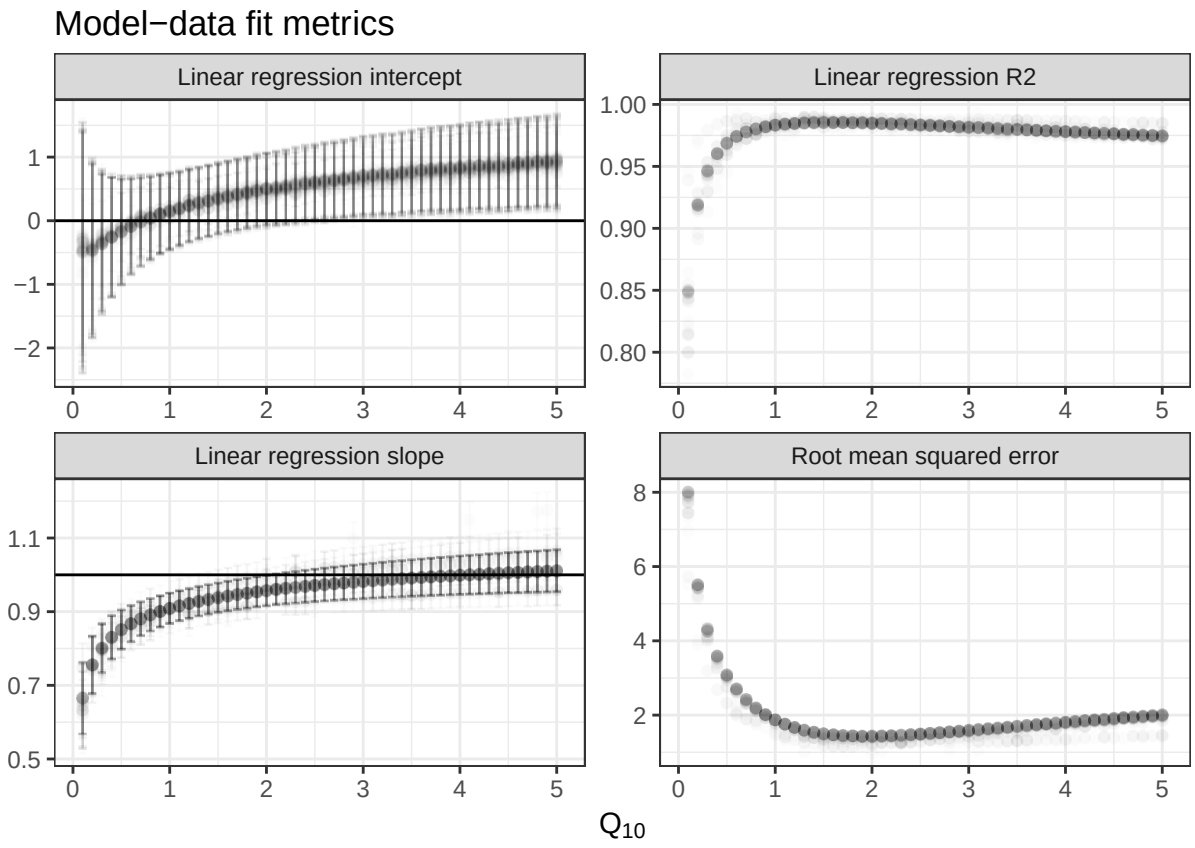
```
  separate(longMetric, into=c('metric', 'type'), sep='\\.', fill='right') %>%
  mutate(type = if_else(is.na(type), 'mean', type)) %>%
  spread(type, value) %>%
  mutate(metric = recode(metric, intercept = 'Linear regression intercept',
         slope = 'Linear regression slope',
         R2 = 'Linear regression R2',
         RMSE = 'Root mean squared error'))

target.df <- data.frame(metric=c('RMSE', 'R2', 'slope', 'intercept'),
                            y=c(NA, NA, 1, 0))%>%
  mutate(metric = recode(metric, intercept = 'Linear regression intercept',
         slope = 'Linear regression slope',
         R2 = 'Linear regression R2',
         RMSE = 'Root mean squared error'))

metric.plot <- ggplot(trueDist, aes(x=Q10)) +
  #geom_ribbon(aes(ymin=min.05, ymax=max.95), color='grey', alpha=0.5) +
  geom_point(aes(y=mean), alpha=0.01) +
  geom_errorbar(aes(ymin=min, ymax=max), alpha=0.01) +
  geom_hline(data=target.df, aes(yintercept=y)) +
  facet_wrap(~metric, scales='free') +
  labs(x=expression(paste(Q[10])), y='', title='Model-data fit metrics') +
  theme_bw()
```

Figure 1: Model-data fit metrics

```
print(metric.plot)
```

```
## Warning: Removed 5000 rows containing missing values (geom_errorbar).
```

```
## Warning: Removed 2 rows containing missing values (geom_hline).
```




```
ggsave(metric.plot, filename='temp/Fig1_modelDataMetrics.pdf')
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 5000 rows containing missing values (geom_errorbar).
```

```
## Warning: Removed 2 rows containing missing values (geom_hline).
```

Figure 1: The median model-data fit metrics (root mean squared error, and linear regression slopes, intercept and R2) across 50 of 1000 random sets of 34 studies. Both the slope and intercept of the linear regression are shown with 2 standard deviations error bars.

Pull good Q10 percentile

```
goodQ10s <- removeSites %>%
  #filter(selectSize>30) %>%
  mutate(goodIntercept = intercept - 2*intercept.sd < 0 &
                         intercept + 2*intercept.sd > 0,
         goodSlope = (slope - 2*slope.sd) < 1 & (slope + 2*slope.sd) > 1)

Q10Summary <-goodQ10s %>% filter(goodIntercept & goodSlope) %>%
  group_by(selectSize, type) %>%
  do( broom::tidy(t(quantile(.$Q10, c(0, 0.05, 0.5, 0.95, 1)))) ) %>%
  rename(`0%`=`X0.`, `5%` = `X5.`, `50%` = `X50.`, `95%` = `X95.`, `100%` = `X100.`)
```

Figure 2: Q10 range across dataset compared with random

```
Q10Range_plot <- ggplot(Q10Summary, aes(x=selectSize, color=type, fill=type)) +
  geom_line(aes(y=`0%`), linetype=2) +
  geom_line(aes(y=`100%`), linetype=2) +
  geom_ribbon(aes(ymin=`5%`, ymax=`95%`), alpha=0.3) +
  geom_line(aes(y=`50%`)) +
  geom_point(aes( y=`50%`)) +
  labs(title=expression(paste('Q'[10],' values with good fits')),
       x=sprintf('Sample size'),
       y=expression(paste(Q[10])),
       color='Sample type',
       fill='Sample type') +
  theme_bw()
ggsave(Q10Range_plot, file='temp/Fig2_Q10Range.pdf', height=4, width=7)
print(Q10Range_plot)
```

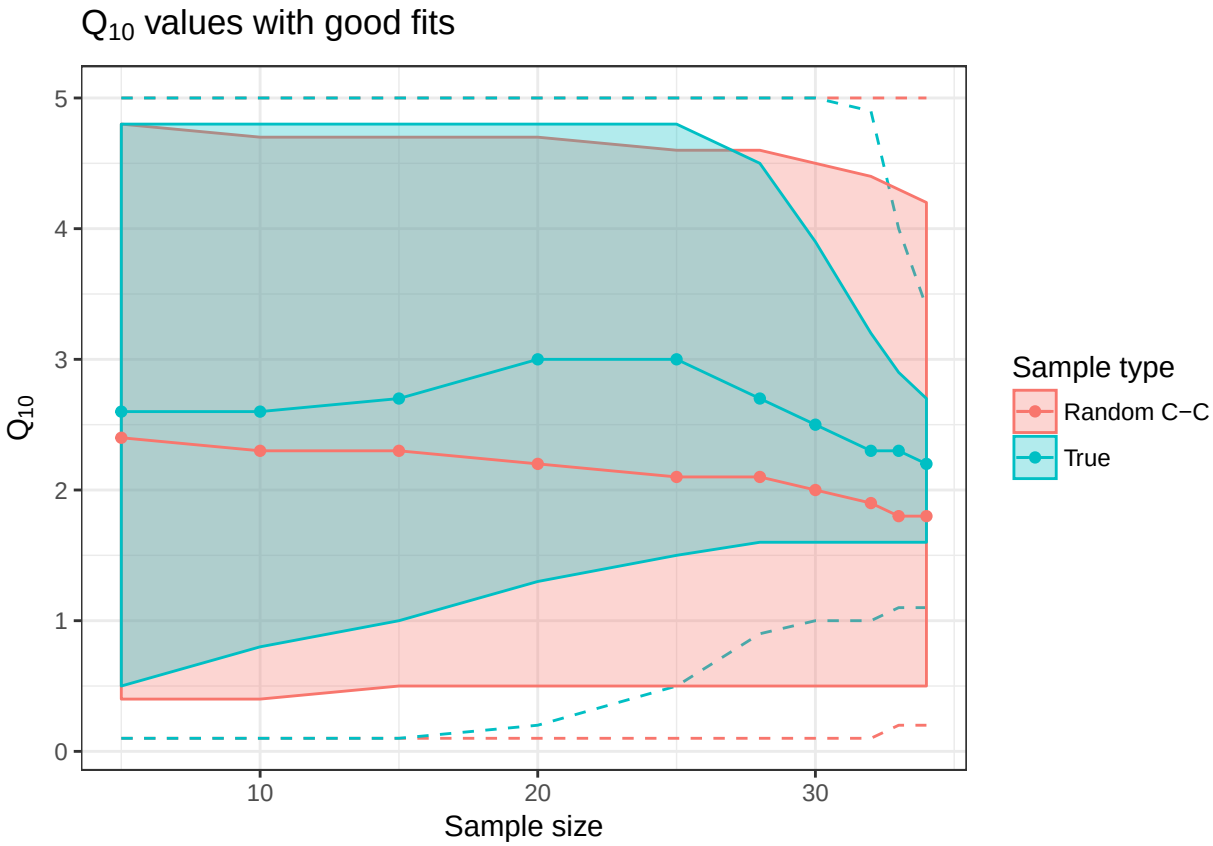


Figure 2: The Q10 with good 1-to-1 fits, at 90% confidence interval (band) with minimum and maximum values (dotted line) and median value (solid line), across 10 different sample sizes ranging from 5 to 34, for the orginal data set (True: blue) and randomized case-control (Random C-C: red).

Figure S1: Null and sample Q10 distribution comparison

```
trueInterval <- removeSites %>%
   mutate(goodIntercept = intercept - 2*intercept.sd < 0 &
```

```
                          intercept + 2*intercept.sd > 0,
        goodSlope = (slope - 2*slope.sd) < 1 & (slope + 2*slope.sd) > 1)

ggplot(subset(trueInterval, goodIntercept & goodSlope)) +
  geom_density(aes(x=Q10, group=selectSize, color=selectSize)) +
  scale_color_gradientn(colors=rainbow(4)) +
  labs(x=expression(paste(Q[10])),
       color='Sample size') +
  facet_wrap(~type)
```

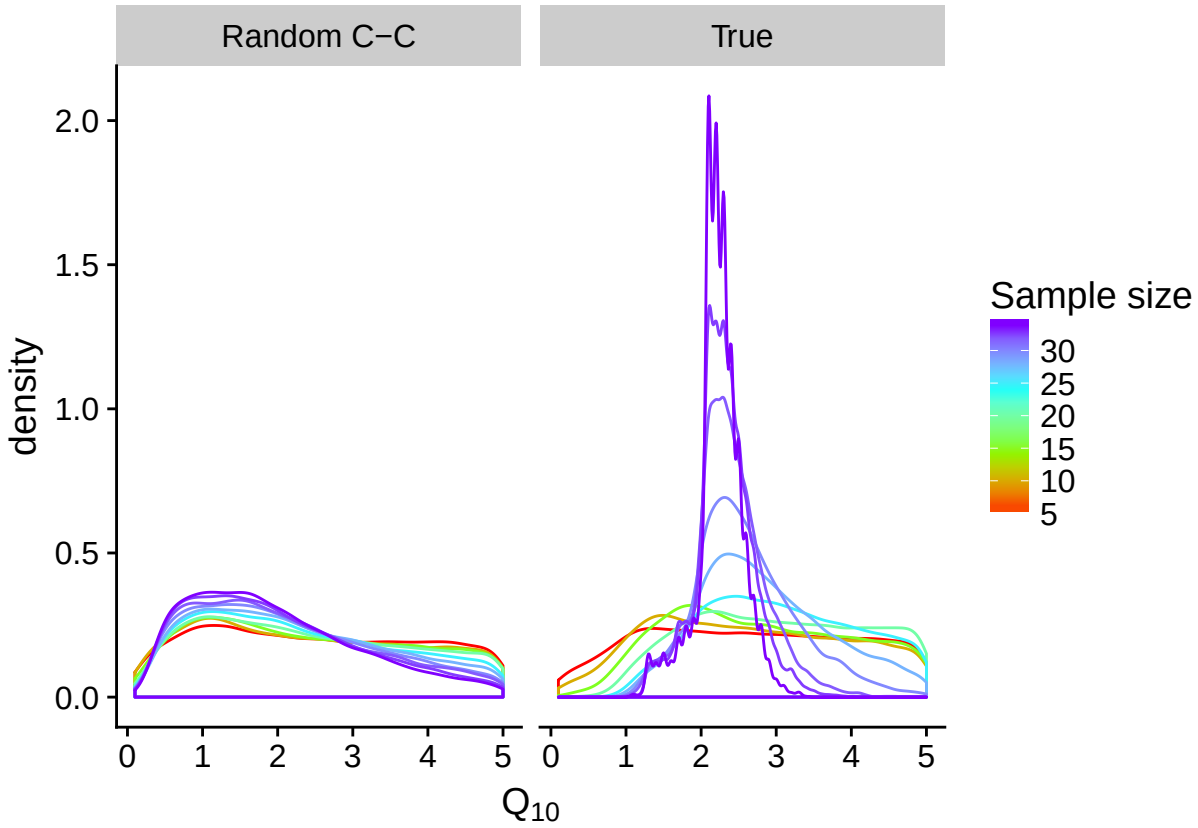


Figure S1: The distribution of Q10's which generate good 1-1 fits for both the orginal true sample and randomized case-control for different sample sizes.

Test that null and sample distributions are distinct

```
testData <- subset(removeSites, intercept - 2*intercept.sd < 0 &
                     intercept + 2*intercept.sd > 0  &
                     (slope - 2*slope.sd) < 1 & (slope + 2*slope.sd) > 1 &
                     selectSize == 34 & type == 'True')$Q10
nullData <- subset(removeSites, intercept - 2*intercept.sd < 0 &
                     intercept + 2*intercept.sd > 0 &
                     (slope - 2*slope.sd) < 1 & (slope + 2*slope.sd) > 1 &
                   selectSize == 34 & type == 'Random C-C')$Q10
print(ks.test( jitter(testData, amount = 0.05), jitter(nullData, amount =0.05)))
```

```
##
```

```
##  Two-sample Kolmogorov-Smirnov test
##
## data:  jitter(testData, amount = 0.05) and jitter(nullData, amount = 0.05)
## D = 0.44095, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

Table S2: Kolmogorv-Smirnov across sample sizes

```
distTest <-  removeSites %>%
   mutate(goodIntercept = intercept - 2*intercept.sd < 0 &
                          intercept + 2*intercept.sd > 0,
          goodSlope = (slope - 2*slope.sd) < 1 & (slope + 2*slope.sd) > 1) %>%
  filter(goodIntercept, goodSlope) %>%
  group_by(selectSize) %>%
  dplyr::summarize(ks_pvalue = ks.test(jitter(Q10[type=='True'], amount = 0.05),
                                       jitter(Q10[type=='Random C-C'],
                                              amount =0.05))$p.value,
                   ks_stat = ks.test(jitter(Q10[type=='True'], amount = 0.05),
                                     jitter(Q10[type=='Random C-C'],
                                            amount = 0.05))$statistic)

distTest.print <- distTest %>% select(-ks_pvalue)
pander(as.data.frame(distTest.print))
```

selectSize	ks_stat
5	0.06082
10	0.1128
15	0.1672
20	0.2334
25	0.3142
28	0.3408
30	0.3644
32	0.3819
33	0.4108
34	0.4415

Table S2: Two-sample Kolmogorv-Smirnov test for different sample sizes. All p-values are below detection (p < 2e-16).

Model-data fit

```
fieldQ10s <- subset(Q10Summary, type=='True' & selectSize == 34, c('5%', '50%', '95%'))

perCw <- data.study %>%
  group_by(Study, Tdelta, Years) %>%
  do((function(xx){
    Q10 <- as.vector(unlist(fieldQ10s))
    return(data.frame(Q10=Q10,
                      data=xx$perC.warmed,
                      data.sd=xx$perC.warmed.sd,
```

```
                    model=xx$perC.control * Q10 ^(-xx$Tdelta/10)))
  })(.))
```

Figure S2: Model-data scatter plot

```
perCw_plot <- ggplot(perCw) +
  geom_errorbarh(aes(y=model, x=data, xmin=data-data.sd, xmax=data + data.sd,
                     color=Q10)) +
  geom_point(aes(y=model, x=data, group=Q10, color=Q10)) +
  geom_abline(slope=1, intercept=0) +
  scale_x_log10() + scale_y_log10() +
  labs(title='Model-data comparison',
       x='Field warmed SOC [mass % organic carbon]',
       y='Modeled warmed SOC [mass % organic carbon]')

ggsave(perCw_plot, filename='temp/fitPlot.pdf')
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 6 rows containing missing values (geom_errorbarh).
```

```
print(perCw_plot)
```

```
## Warning: Removed 6 rows containing missing values (geom_errorbarh).
```

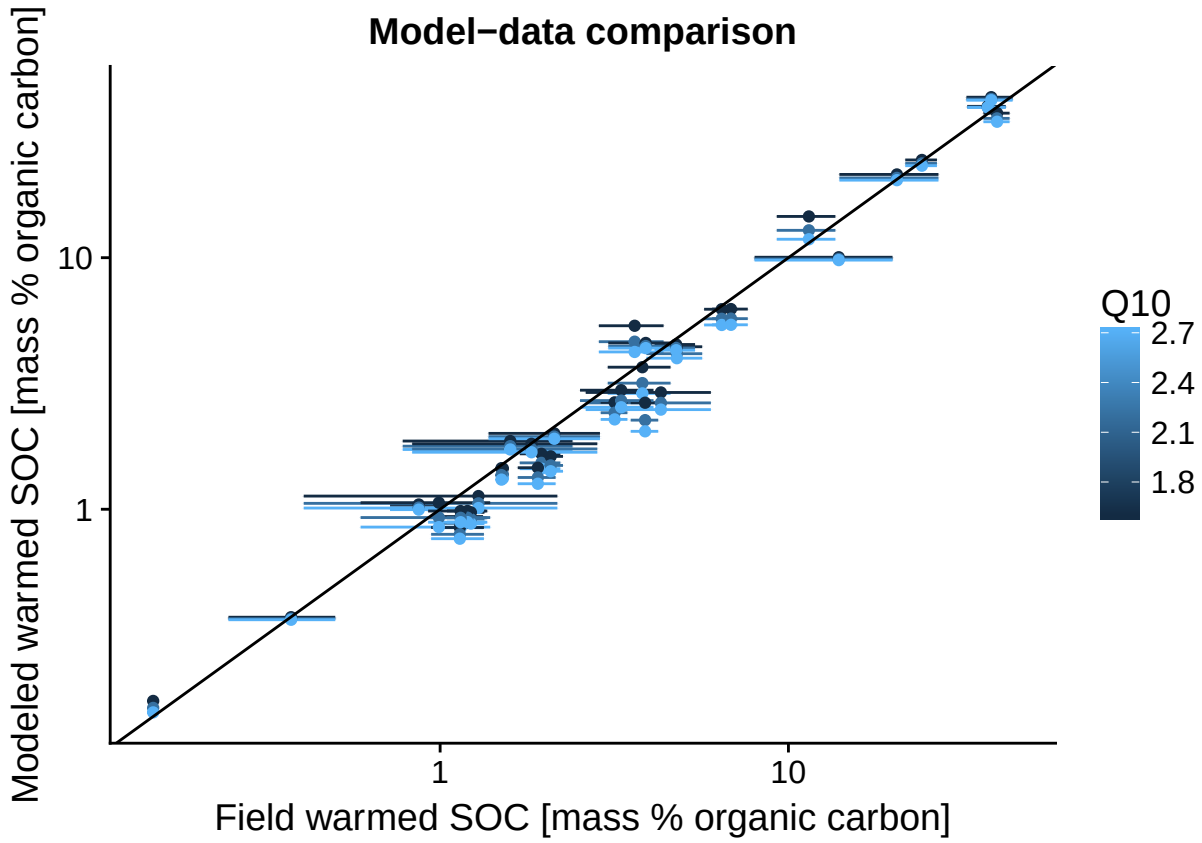


Figure S2: Model data comparison for the 5%, 50%, 95% percentiles of valid Q10 range for 34 random subsets of the full data set (Q10 values = []). 1-to-1 line provided for reference.

Earth system model constraints

Table S3: CMIP5 ESM output

Prepare CMIP5 files

```
cmip5dir <- '/Volumes/PNNL_ToddBrown/CMIP5/rcp85'

local.cmip5dir <- '~/Documents/CurrentProjects/soilFieldWarmingMeta/cmip5'
for(varID in c('areacella', 'sftlf', 'rh', 'cSoil', 'cLitter', 'tsl')){
  if(!file.exists(file.path(local.cmip5dir, varID))){
    dir.create(file.path(local.cmip5dir, varID))
  }
}

modelCenter.df <- readxl::read_excel('data/CMIP5names.xlsx', sheet=1, trim_ws=TRUE) %>%
  mutate_all(function(xx){gsub('\\s*$', '', gsub('^\\s*', '', as.character(xx)))})


allOutputs <- data.frame(filename=list.files(path=cmip5dir,
                                             pattern='nc', full.names=TRUE,
                                             recursive=TRUE),
                         stringsAsFactors = FALSE) %>%
  mutate(filebase = gsub('.nc$', '', gsub("^.*/", '', filename)),
         path=gsub('/[^/]*$', '', filename),
         size = file.size(filename)) %>%
  mutate(filename=filebase) %>%
  filter(size > 0) %>%
  separate(filebase,
           into=c('variable', 'domain', 'model', 'experiment', 'ensemble', 'time'),
           sep='_') %>%
  filter(experiment == 'rcp85') %>%
  group_by(model) %>%
  mutate(hasAllVars = all(c('areacella', 'sftlf', 'cSoil', 'rh', 'tsl') %in%
                            variable)) %>%
  filter(hasAllVars, variable %in%
           #c('areacella', 'sftlf', 'cSoil', 'rh', 'tsl', 'cLitter', 'cCwd'))
           c('areacella', 'sftlf', 'cSoil', 'rh', 'tsl', 'cLitter')) #remove cCwd for 2018 April bgs re

outputFiles <- allOutputs %>%
  filter(grepl('(^20)|^\\s*$', time)) %>%
  #convert string years to numeric YYYYMM Works because everything is 21st century
  mutate(startTime=as.numeric(substr(time, 1, 6)),
         endTime=as.numeric(substr(time, 8, 13))) %>%
  group_by(domain, experiment, model, variable, ensemble) %>%
  #update new files to point to future merge files
  summarize(path2=ifelse(length(path) == 1, path, sprintf('%s/temp', cmip5dir)),
                  filename2 = ifelse(length(filename) == 1, filename,
                                     sprintf('%s_%s_%s_rcp85_%s_%d-%d',
                                             variable[1], domain[1],
                                             model[1], ensemble[1],
                                             min(startTime),
                                             max(endTime))),
```

Modeling Center (or Group)	Institute ID	Model	Variable	Time (YYYYMM)	ensemble
Beijing Climate Center, China Meteorological Administration	BCC	BCC-CSM1.1	cLitter	200601-209912	r1i1p1
			cSoil		
			rh		
			tsl		
Canadian Centre for Climate Modelling and Analysis	CCCma	CanESM2	cLitter	200601-210012	r1i1p1;r2i1p1;r3i1p1;r4i1p1;r5i1p1
			cSoil		
			rh		
			tsl		
National Center for Atmospheric Research	NCAR	CCSM4	cSoil		r1i1p1;r2i1p1;r3i1p1;r4i1p1;r5i1p1;r6i1p1
			rh		
			cCwd		
			cLitter		
			tsl		
Community Earth System Model Contributors (National Science Foundation, Department of Energy, National Center for Atmospheric Research)	NSF-DOE-NCAR	CESM1(BGC)	cCwd		r1i1p1
			cLitter		
			rh		
			tsl		
			cSoil		
		CESM1(CAM5)	cCwd		r1i1p1;r2i1p1;r3i1p1
			cLitter		
			rh		
			tsl		
			cSoil		
		CESM1(WACCM)	cCwd	200601-209912	r3i1p1;r4i1p1
			cSoil		
			rh		
			tsl		
			cLitter		
NOAA Geophysical Fluid Dynamics Laboratory	NOAA GFDL	GFDL-ESM2G	rh	200601-210012	r1i1p1
			cSoil		
			tsl		
		GFDL-ESM2M	cSoil		
			rh		
			tsl		
Met Office Hadley Centre (*additional HadGEM2-ES realizations contributed by Instituto Nacional de Pesquisas Espaciais)	MOHC*	HadGEM2-CC	cSoil	200512-209912	r1i1p1;r2i1p1;r3i1p1
			tsl		
			rh		
		HadGEM2-ES	rh	200512-210011	r2i1p1;r3i1p1;r4i1p1
			tsl		
			cSoil		
Institute for Numerical Mathematics	INM	INM-CM4	tsl	200601-210012	r1i1p1
			cSoil		
			rh		
Institut Pierre-Simon Laplace	IPSL	IPSL-CM5A-LR	cLitter		r2i1p1;r3i1p1;r4i1p1
			tsl		
			cSoil		
			rh		
		IPSL-CM5A-MR	cSoil		r1i1p1
			cLitter		
			rh		
			tsl		
		IPSL-CM5B-LR	cSoil		
			cLitter		
			rh		
			tsl		
Japan Agency for Marine-Earth Science and Technology, Atmosphere and Ocean Research Institute (The University of Tokyo), and National Institute for Environmental Studies	MIROC	MIROC-ESM	cLitter		
			cSoil		
			rh		
			tsl		
		MIROC-ESM-CHEM	cSoil		
			rh		
			cLitter		
			tsl		
Max Planck Institute for Meteorology (MPI-M)	MPI-M	MPI-ESM-MR	cLitter		
			cSoil		
			rh		
			tsl		
Meteorological Research Institute	MRI	MRI-ESM1	cLitter		
			cSoil		
			rh		
			tsl		
Norwegian Climate Centre	NCC	NorESM1-M	cCwd		
			cLitter		
			cSoil		
			rh		
			tsl		
		NorESM1-ME	cCwd		
			cSoil		
			rh		
			tsl		
			cLitter		

Figure 1: Table S3: CMIP5 Earth system model output information.

```
                  time2=ifelse(length(time) == 1, time,
                               sprintf('%d-%d', min(as.numeric(startTime)),
                                       max(as.numeric(endTime)))),
                  startTime2=min(as.numeric(startTime)),
                  endTime2=max(as.numeric(endTime)),
                  mergeTimeFilesFlag=length(filename) > 1) %>%
  rename(path=path2, filename=filename2,
                time=time2, startTime=startTime2, endTime=endTime2) %>%
  #Construct system call to merge files
  mutate(NCO_mergeTime=ifelse(mergeTimeFilesFlag,
                              sprintf('ncrcat %s/%s_%s_%s_rcp85_%s_20*.nc %s/%s.nc',
                                      path, variable, domain, model, ensemble,
                                      path, filename), '')) %>%
  ##Filter outputs to ensure they cover 2007, 2099
  filter(startTime <= 200700 & endTime >= 209900) %>%
  ##pull month-year from times
  mutate(startYear = floor(startTime/100), endYear = floor(endTime/100),
         startMonth = startTime %% 100, endMonth = endTime %% 100)

##Merge time slices
outputFiles %>%
  group_by(NCO_mergeTime) %>%
  do((function(xx){
    if(!file.exists(sprintf('%s/%s.nc',xx$path, xx$filename))){
      cat('-')
      cat(xx$NCO_mergeTime)
      system(xx$NCO_mergeTime)
      cat('+')
      ans <- data.frame(status='executed')
    }else{
      ans <- data.frame(status='not executed')
    }
    return(ans)
  })(.))
```

```
## # A tibble: 50 x 2
## # Groups:   NCO_mergeTime [50]
##    NCO_mergeTime                                                  status
##    <chr>                                                          <fct>
##  1 ""                                                             not ex~
##  2 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cLitter_Lmon_C~ not ex~
##  3 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cLitter_Lmon_C~ not ex~
##  4 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cLitter_Lmon_N~ not ex~
##  5 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cSoil_Lmon_CES~ not ex~
##  6 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cSoil_Lmon_CES~ not ex~
##  7 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cSoil_Lmon_GFD~ not ex~
##  8 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cSoil_Lmon_GFD~ not ex~
##  9 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cSoil_Lmon_Had~ not ex~
## 10 ncrcat /Volumes/PNNL_ToddBrown/CMIP5/rcp85/temp/cSoil_Lmon_Had~ not ex~
## # ... with 40 more rows
```

```
finalOutputs <- outputFiles %>%
  group_by(model, time) %>%
  ##trim non-standard timespans; remove odd ensemble out
```

```
  select(model, time, ensemble) %>%
  dplyr::summarize(numRep = length(unique(ensemble))) %>%
  group_by(model) %>%
  dplyr::summarize(refTime=time[which.max(numRep)]) %>%
  dplyr::rename(time=refTime) %>%
  left_join(outputFiles, by=c('model', 'time')) %>%
  ##set up NCO command to slice modern and future years
  group_by(domain, experiment, model, variable, time,
           startYear, startMonth, endYear, endMonth) %>%
  dplyr::summarize(
    inputFiles = paste0(sprintf('%s/%s.nc', path, filename), collapse=' '),
    NCO_mergeTime_hist = paste0(NCO_mergeTime[grepl('n', NCO_mergeTime)], collapse=';'),
    ensemble_ls = paste0(ensemble, collapse=';')) %>%
  mutate(path=local.cmip5dir,
         ##Account for Dec 2005 start date; the only starting years are 2005 and 2006
         startIndex_offset = ((13-startMonth) %% 12)) %>%
  group_by(path, domain, experiment, model, variable, time,
           inputFiles, NCO_mergeTime_hist, ensemble_ls,
           startYear, startMonth, endYear, endMonth, startIndex_offset) %>%
  do((function(.){
    return(data.frame(
      newStartYear = c(2007, 2011, 2016, 2021, 2026, 2056, 2081, 2006, 2089),
      newEndYear   = c(2007, 2011, 2016, 2021, 2026, 2056, 2081, 2006, 2089) + 10
    ))})(.)) %>%
  mutate(outfilename =
           if_else(newStartYear == newEndYear,
                   sprintf('%s/%s_%s_%s_%s_%s_%d', variable, variable, domain, model,
                           experiment, newStartYear),
                   sprintf('%s/%s_%s_%s_%s_%s_%d-%d', variable, variable, domain, model,
                           experiment, newStartYear, newEndYear)),
         startIndex = startIndex_offset + 12*(newStartYear - 2006),
         countIndex   = 12*(newEndYear - newStartYear + 1) - 1 )%>%
  mutate(endIndex = startIndex + countIndex) %>%
  mutate(NCO_mergeEnsem = sprintf('nces -d time,%d,%d %s -o %s/%s.nc',
                                  startIndex, endIndex,
                                  inputFiles, path, outfilename))

  ##Execute the emseble merge and time slice
  finalOutputs %>%
  group_by(NCO_mergeEnsem) %>%
    do((function(path=.$path, outfile=.$outfilename, cmd=.$NCO_mergeEnsem){
    if(!file.exists(sprintf('%s/%s.nc', path, outfile))){
      print(cmd)
      system(cmd)
    }
    return(data.frame())
  })())
```

```
## # A tibble: 0 x 1
## # Groups:   NCO_mergeEnsem [0]
## # ... with 1 variable: NCO_mergeEnsem <chr>
```

```
fixOutputs <- allOutputs %>%
  filter(variable %in% c('areacella', 'sftlf')) %>%
```

```
  dplyr::rename(ensemble_ls=ensemble)

finalOutputs <- finalOutputs %>% bind_rows(fixOutputs)
```

Calculate gridded anaual averages from monthly values

```
outputFileDir <- '~/Documents/CurrentProjects/soilFieldWarmingMeta/cmip5/RData/'
if(!file.exists(outputFileDir)){
  dir.create(outputFileDir)
}

modelOutputs <- finalOutputs %>%
  group_by(model, newStartYear, newEndYear) %>%
  do((function(xx){
    if(is.na(xx$newStartYear)[1]){
      #go through timeless in fixOutputs
      saveFile <- paste0(outputFileDir, xx$model[1], '_fixed.RData')
      maps <- list()

      if(!file.exists(saveFile)){
        for(varID in xx$variable){
          temp.nc <- nc_open(paste0(xx$path[xx$variable == varID], '/',
                                    xx$filename[xx$variable == varID], '.nc'))
          maps[[varID]] <- ncvar_get(temp.nc, varid=varID)
          nc_close(temp.nc)
        }
        save(maps, file=saveFile)
        rm(maps)
      }
      return(data.frame(filename=saveFile))
    }##End na start year if


    numMonths <- (xx$newEndYear[1] - xx$newStartYear[1])*12
    if(xx$newEndYear[1] == 2099){
      yrSpan <- 2099 - c(numMonths/12 -1, 0)
    }else{
      yrSpan <- xx$newStartYear[1]+c(0, numMonths/12 - 1)
    }
    saveFile <-  paste0(outputFileDir, xx$model[1], '_',
                        yrSpan[1], '-', yrSpan[2],'.RData')
    if(!file.exists(saveFile)){
      maps <- list(SOC = 0)
      #saveFile <- paste0(outputFileDir, xx$model[1], '_', xx$newStartY '.RData')
      #remove cCwd for April 2018 BGS review
      for(varID in intersect(c('cSoil', 'cLitter', 'rh', 'tsl'), xx$variable)){
        temp.nc <- nc_open(paste0(xx$path[xx$variable == varID], '/',
                                  xx$outfilename[xx$variable == varID], '.nc'))
        if(varID %in% c('cSoil', 'cLitter')){#, 'cCwd')){  #remove cCwd for April 2018 BGS review
          maps$SOC <- maps$SOC + ncvar_get(temp.nc, varid=varID)
        }else{
          maps[[varID]] <- ncvar_get(temp.nc, varid=varID)
```

```
        }

        if(varID == 'tsl'){
          #cut depth to 10 cm
          depth <- ncvar_get(temp.nc, varid='depth_bnds')
          depth[depth > 0.1] <- 0.1
          depthWeight <- (depth[2,] - depth[1,])/0.1
          #average top 10cm and convert from K to C
          maps$tsl <- apply(maps$tsl, c(1,2,4),
                            function(xx){sum(xx*depthWeight) - 273.15})
        }else if(varID == 'rh'){
          maps$rh <- maps$rh * 3.154e+7 #convert from seconds to years
        }
        nc_close(temp.nc)
      }
      #construct monthly change in soil organic carbon
      maps$dSOC <- (maps$SOC[,,-1] - maps$SOC[,,-dim(maps$SOC)[3]])/12

      maps <- lapply(maps, function(yy){
        apply(yy[,,ifelse(xx$newEndYear[1] == 2099,
                          dim(yy)[3] - 0:(numMonths-1),
                          1:numMonths)], c(1,2), mean)})

      save(maps, file=saveFile)
      rm(maps)
    }
    return(data.frame(filename=saveFile, numMonths=numMonths,
                      yrStr=paste0(yrSpan[1], '-', yrSpan[2])))
  })(.))
```

Pull all gridded variables together

```
#ggplot(filter(Q10Map, is.finite(value))) + geom_histogram(aes(x=value))
dC.maps <- data.frame(startYrStr=c('2006-2015', '2006-2015', '2006-2015',
                                   '2006-2015', '2006-2015', '2006-2015',
                                   '2006-2015', '2006-2015'),
             endYrStr=c('2007-2016', '2011-2020', '2016-2025',
                        '2021-2030', '2026-2035', '2056-2065',
                        '2081-2090','2090-2099')) %>%
  left_join(modelOutputs[,c('model', 'yrStr')], by=c('startYrStr' = 'yrStr')) %>%
  group_by(model, startYrStr, endYrStr) %>%
  do((function(xx){
    #print(paste(xx$model, xx$startYrStr, xx$endYrStr))
    startID <- modelOutputs %>% filter(as.character(xx$model) == model,
                                       yrStr == xx$startYrStr)
    load(startID$filename)
    start.ls <- maps
    start.ls$input <- start.ls$rh - start.ls$dSOC

    endID <- modelOutputs %>% filter(as.character(xx$model) == model,
                                     yrStr == xx$endYrStr)
    load(endID$filename)
```

```
    end.ls <- maps
    end.ls$input <- end.ls$rh - end.ls$dSOC

    fixedID <- modelOutputs %>% filter(as.character(xx$model) == model, is.na(yrStr))
    load(fixedID$filename)
    fixed.ls <- maps

    ans <- data.frame(reshape2::melt(end.ls$tsl - start.ls$tsl,
                                     varnames=c('lon', 'lat'), value.name='dT'),
                      SOC.start = reshape2::melt(start.ls$SOC)$value,
                      SOC.end = reshape2::melt(end.ls$SOC)$value,
                      input.start=reshape2::melt(start.ls$input)$value,
                      input.end=reshape2::melt(end.ls$input)$value,
                      rh.start=reshape2::melt(start.ls$rh)$value,
                      rh.end=reshape2::melt(end.ls$rh)$value,
                      tsl.start=reshape2::melt(start.ls$tsl)$value,
                      tsl.end=reshape2::melt(end.ls$tsl)$value,
                      landArea = reshape2::melt(fixed.ls$sftlf/100*
                                                  fixed.ls$areacella)$value) %>%
      filter(is.finite(dT+input.start+input.end+rh.start+rh.end+SOC.start+SOC.end))

    ans$lon <- ans$lon/nrow(start.ls$SOC)*360-180
    ans$lat <- ans$lat/ncol(start.ls$SOC)*180-90

    return(ans)
  })(.)) %>%
  mutate(gapYrs = as.numeric(substr(endYrStr, 1, 4)) -
           as.numeric(substr(startYrStr, 1, 4)))
```

```
## Warning: Column `startYrStr`/`yrStr` joining factor and character vector,
## coercing into character vector
```

Figure S3: Input-output balance of soils in Earth system models

In general, most grid cells experience less then a 1% shift in their soil carbon stocks, in most cases under 0.1%. This implies that the grid cells are at quasi-steady state where the inputs approximately equals the output. That is that, to a first order approximation, the inputs roughly equal the outputs.

```
quantileArr <- c(0, 0.5, 1, 2.5, 5, 7, 10, 50, 90, 93, 95, 97.5, 98, 99, 99.5, 100)
relTol <- 1e-3
ss.ranges <- dC.maps %>%
  ungroup() %>% group_by(model, startYrStr) %>%
  select(model, startYrStr, lat, lon, ends_with('start')) %>%
  rename(yrStr=startYrStr) %>%
  unique() %>%
  mutate(input.start = if_else(input.start < max(abs(input.start))*relTol,
                               0, input.start),
         rh.start = if_else(rh.start < max(abs(rh.start))*relTol,
                            0, rh.start)) %>%
  summarize( quantile.lab = list(sprintf("X%1.1f", quantileArr)),
             balance = list(quantile(abs(input.start-rh.start)/
                                       pmax(input.start, rh.start), quantileArr/100,
                                     na.rm=TRUE))) %>%
```

```
  unnest() %>%
  spread(key=quantile.lab, value=balance) %>%
  bind_rows(dC.maps %>%
              filter(endYrStr == '2090-2099') %>%
              ungroup() %>% group_by(model, endYrStr) %>%
              select(model, lat, lon, endYrStr, ends_with('end')) %>%
              rename(yrStr=endYrStr) %>%
              unique() %>%
              mutate(input.end = if_else(input.end < max(abs(input.end))*relTol,
                                         0, input.end),
                     rh.end = if_else(rh.end < max(abs(rh.end))*relTol, 0, rh.end)) %>%
              summarize(quantile.lab = list(sprintf("X%1.1f", quantileArr)) ,
                        balance = list(quantile(abs(input.end-rh.end)/
                                                  pmax(input.end, rh.end),
                                                quantileArr/100, na.rm=TRUE))) %>%
              unnest() %>%
              spread(key=quantile.lab, value=balance))
```

```
## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector
```

```
ggplot(ss.ranges) +
  geom_boxplot(aes(x=model, ymin=`X2.5`, lower=`X7.0`, middle=`X50.0`,
                   upper=`X93.0`, ymax=`X98.0`), stat='identity') +
  #geom_point(aes(x=model, y=`X0.0`)) +
  geom_point(aes(x=model, y=`X100.0`)) +
  scale_y_log10(limits=c(1e-5, 1)) +
  geom_hline(yintercept=0.1, linetype=2) +
  labs(y='Relative SOC imbalance') +
  facet_wrap(~yrStr) +
  theme(axis.text.x=element_text(angle=90, vjust=0.5, hjust=1),
        axis.title.x=element_blank())
```

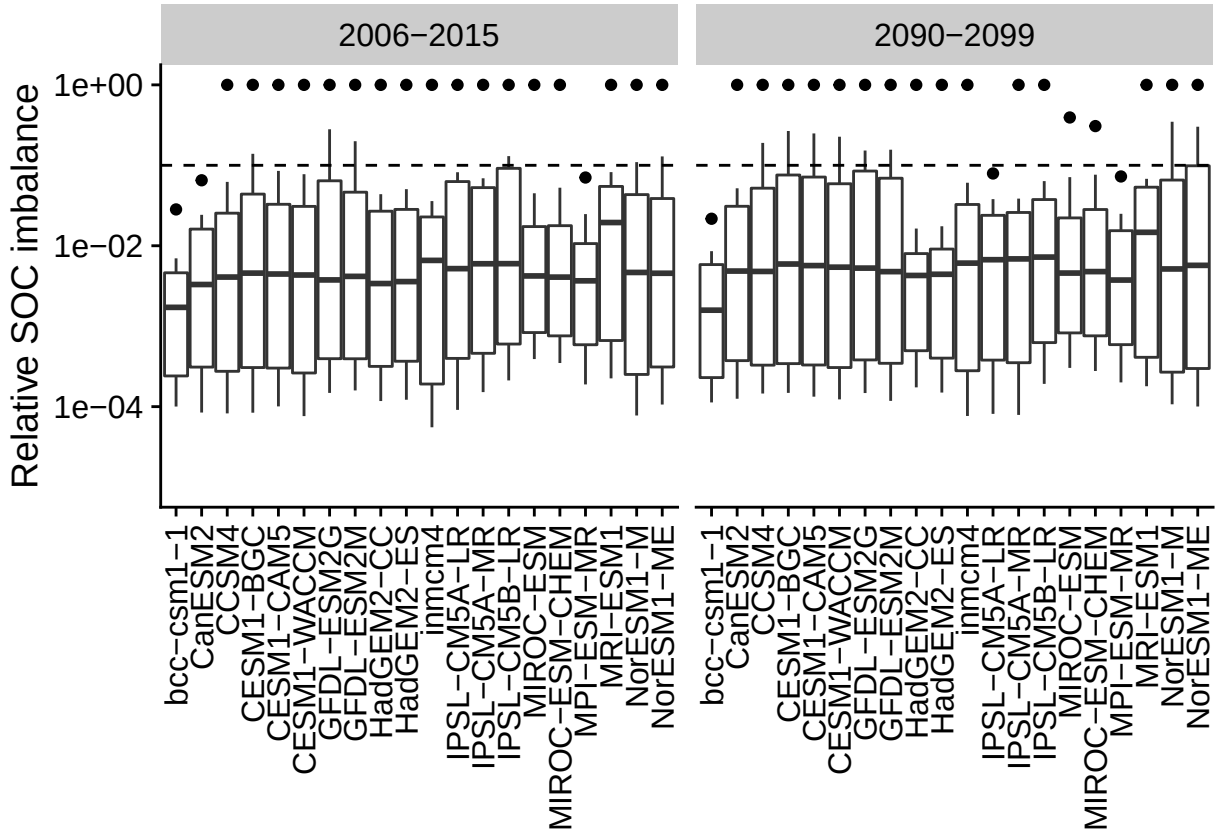


Figure S3: The input-output balance of soils in different Earth system models. Bar represents the median grid value for each model, boxes the 7-93% range, whiskers the 95% interval, and dots the maximum. The dashed line represent the range at which the outputs are within 10% of the inputs.

Calculate a gridded Q10 value from change in SOC

```
#Formula derivation:
#Do not assume quasi steady state
#Rh = SOC*k*Q10^((T-T0)/10)
#rh.end/rh.start = soc.end/soc.start * Q10^((T.end-T0)) / Q10^((T.start-T0)/10)
#                = soc.end/soc.start * Q10 ((T.end-T.start)/10)
#rh.end/rh.start = soc.end/soc.start * Q10 ^(dT/10)
#dT/10 * log(Q10) = log(rh.end/rh.start * soc.start/soc.end)
#
#Assume quasi steady state
#SOC=input/(k*Q10^((T-T0)/10))
#SOC.end/SOC.start=input.end/input.start * Q10 ^ ((T.start-T.end)/10)
#                 =inputRatio * Q10 ^ (-dT/10)
#(-dT/10) * log(Q10) = log(SOC.end/SOC.start * input.start / input.end)
#                    = log(SOC.end) - log(SOC.start) + log(input.start) - log(input.end)
# dT/10 * log(Q10) = log(SOC.start) - log(SOC.end) + log(input.end) - log(input.start)
# dT/10 * log(Q10) = log(SOC.start/SOC.end*input.end/input.start)
dC.maps <- dC.maps %>%
  mutate(Q10.dRh = exp(10/dT * log(rh.end/rh.start*SOC.start/SOC.end)),
         #Assume one pool approximation
         Q10.dIn = exp(10/dT * log(input.end/input.start*SOC.start/SOC.end))) %>%
```

```
  #Assume one pool approx and quasi steady state
  left_join(modelCenter.df %>% select(center, model, land.model,
                                      temperature.model, model.name), by='model')
```

Q10 over different simulation times

Figure S4: Compare annual Q10 distribution across different time steps

```
RH_plot <- ggplot(dC.maps %>%
  filter(is.finite(Q10.dRh), Q10.dRh > 0.5, Q10.dRh < 5)) +
  geom_density(aes(x=Q10.dRh, group=gapYrs, color=gapYrs)) +
  scale_color_gradientn(trans = "log10", colors=rainbow(4)) +
  labs(title='Gridded Q10 values from Rh',
       color='Time (yrs)', x='Temperature sensitivity (Q10)') +
  guides(color=FALSE) +
  facet_wrap(~model.name)

In_plot <- ggplot(dC.maps %>%
  filter(is.finite(Q10.dIn), Q10.dIn > 0.5, Q10.dIn < 5)) +
  geom_density(aes(x=Q10.dIn, group=gapYrs, color=gapYrs)) +
   scale_color_gradientn(trans = "log10", colors=rainbow(4)) +
  labs(title='Gridded Q10 values from inputs',
       color='Time (yrs)', x='Temperature sensitivity (Q10)') +
  facet_wrap(~model.name)

plot_grid(RH_plot, In_plot, labels = c("A", "B"), nrow=2)
```

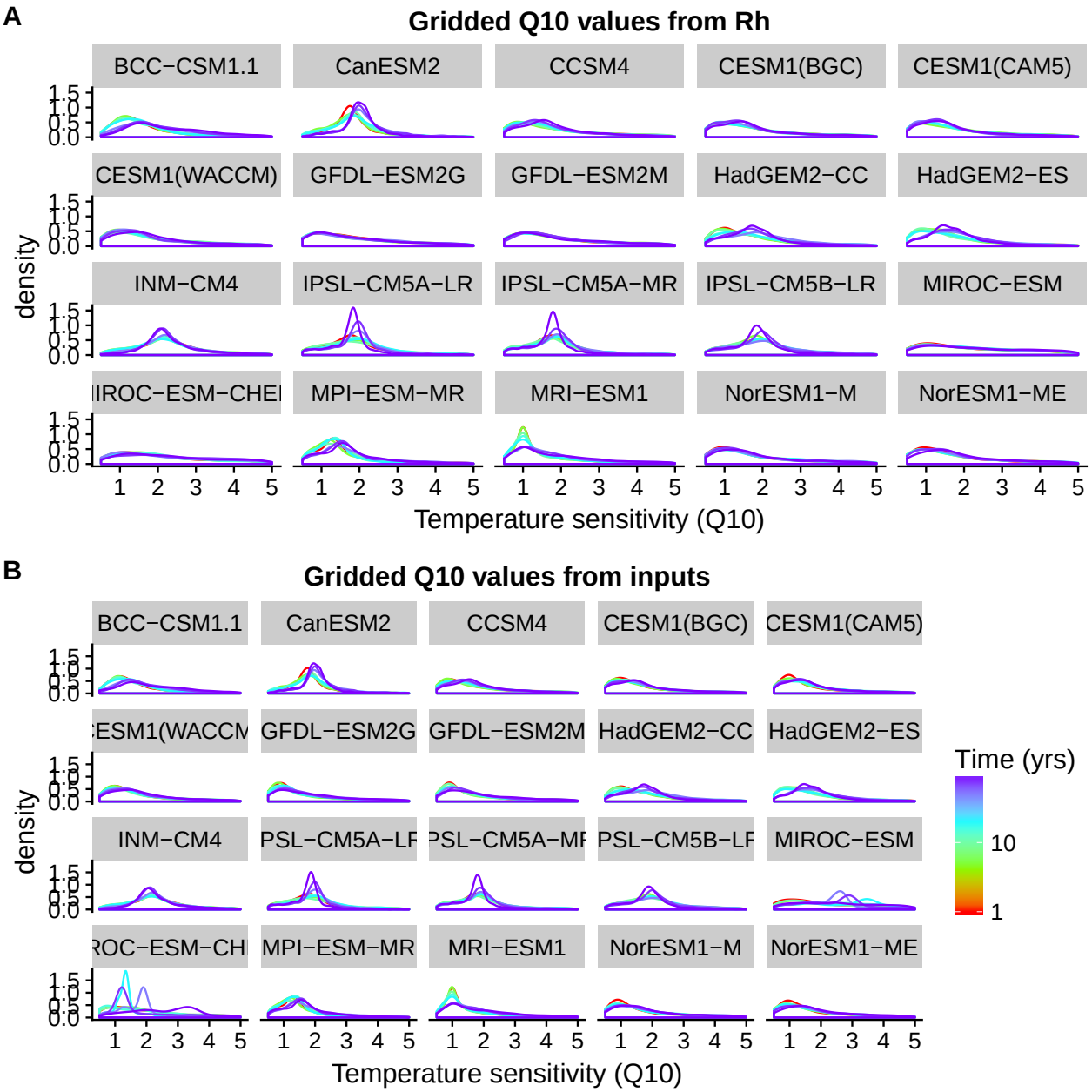


Figure S4: Distribution of gridded Q10 values infered from A) the one pool approximation using hetertrophic respiration (outflux) and B) a one pool approximation combined with a metastable state assumption thus using soil inputs (influx) instead of outflux. Values taken from 10 year means with varying gaps from 1 to 84 years, denoted by colors.

Figure S5: Variation in Q10 distribution within climate centers

```
RH_plot <- ggplot(dC.maps %>%
  filter(is.finite(Q10.dRh), Q10.dRh > 0.5, Q10.dRh < 5) %>%
  filter(startYrStr == '2006-2015', endYrStr == '2090-2099')) +
  geom_density(aes(x=Q10.dRh, group=model, color=temperature.model)) +
  labs(title='Gridded Q10 values over 21st century', x='Q10 (Rh)') +
  guides(color=FALSE) +
```

```
  facet_wrap(~land.model)

In_plot <- ggplot(dC.maps %>%
  filter(is.finite(Q10.dIn), Q10.dIn > 0.5, Q10.dIn < 5) %>%
  filter(startYrStr == '2006-2015', endYrStr == '2090-2099')) +
  geom_density(aes(x=Q10.dIn, group=model, color=temperature.model)) +
  labs(title='Gridded Q10 values over 21st century', x='Q10 (Input)') +
  facet_wrap(~land.model)

plot_grid(RH_plot, In_plot, labels = c("A", "B"), nrow=2)
```

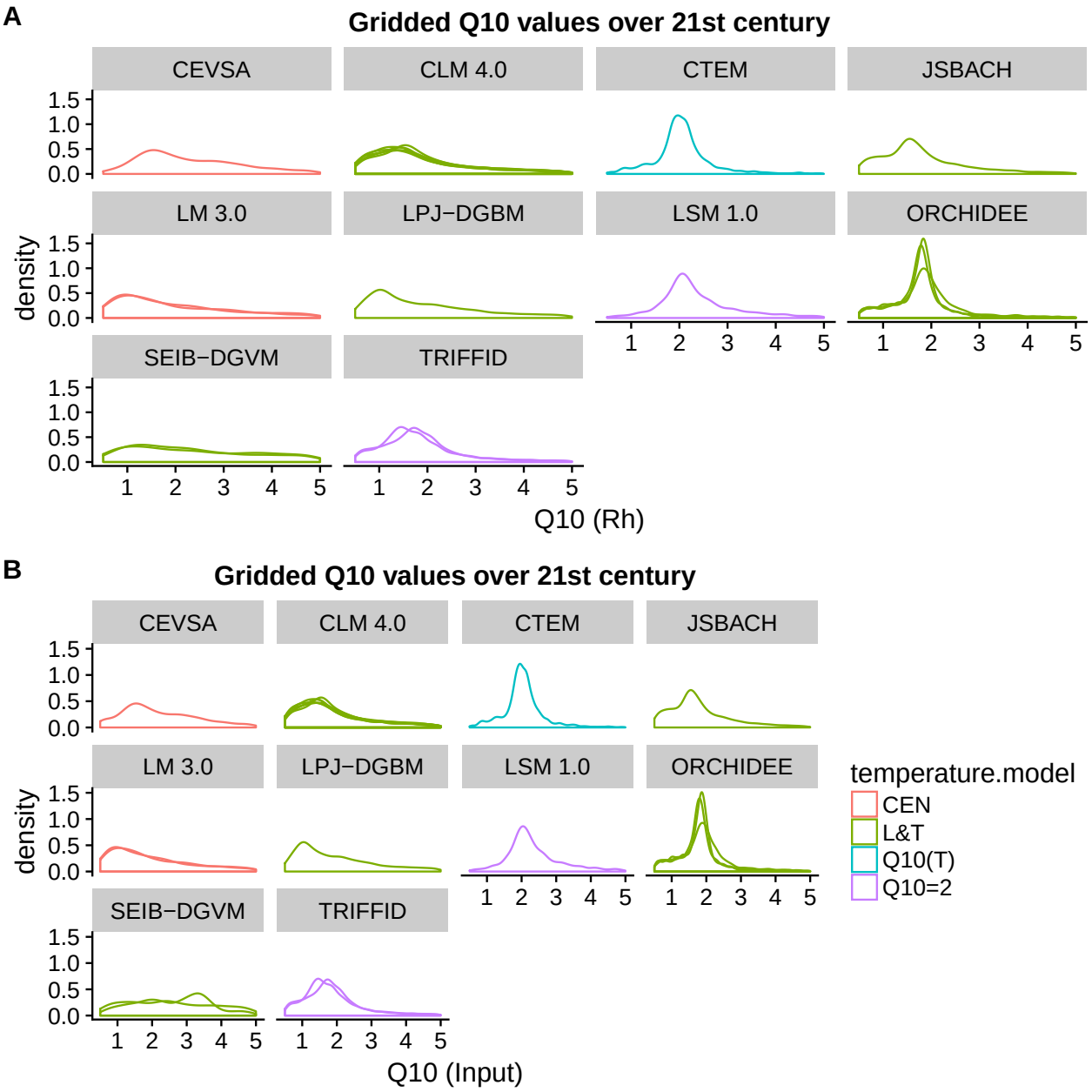


Figure S5: Distribution of gridded Q10 values by land model calculated from the 10yr means at the begining and end of the 21st century using A) the one pool approximation using hetertrophic respiration (outflux) and B) a one pool approximation combined with a metastable state assumption thus using soil inputs (influx)

instead of outflux. Temperature models are coded as colors (Cen=Century, L&T = Lolyd and Taylor, other keys are explained in Table 1 in the main paper).

Figure S6: Metastable assumption and Q10

```
ggplot(dC.maps %>%
  filter(is.finite(Q10.dIn), Q10.dIn > 0.5, Q10.dIn < 5) %>%
  filter(is.finite(Q10.dRh), Q10.dRh > 0.5, Q10.dRh < 5) %>%
  filter(startYrStr == '2006-2015', endYrStr == '2090-2099')) +
  geom_point(aes(x=Q10.dRh, y=Q10.dIn), alpha=0.1) +
  labs(title='Gridded Q10 values over 21st century', x='Q10 (Rh)', y='Q10 (input)') +
  facet_wrap(~model.name)
```

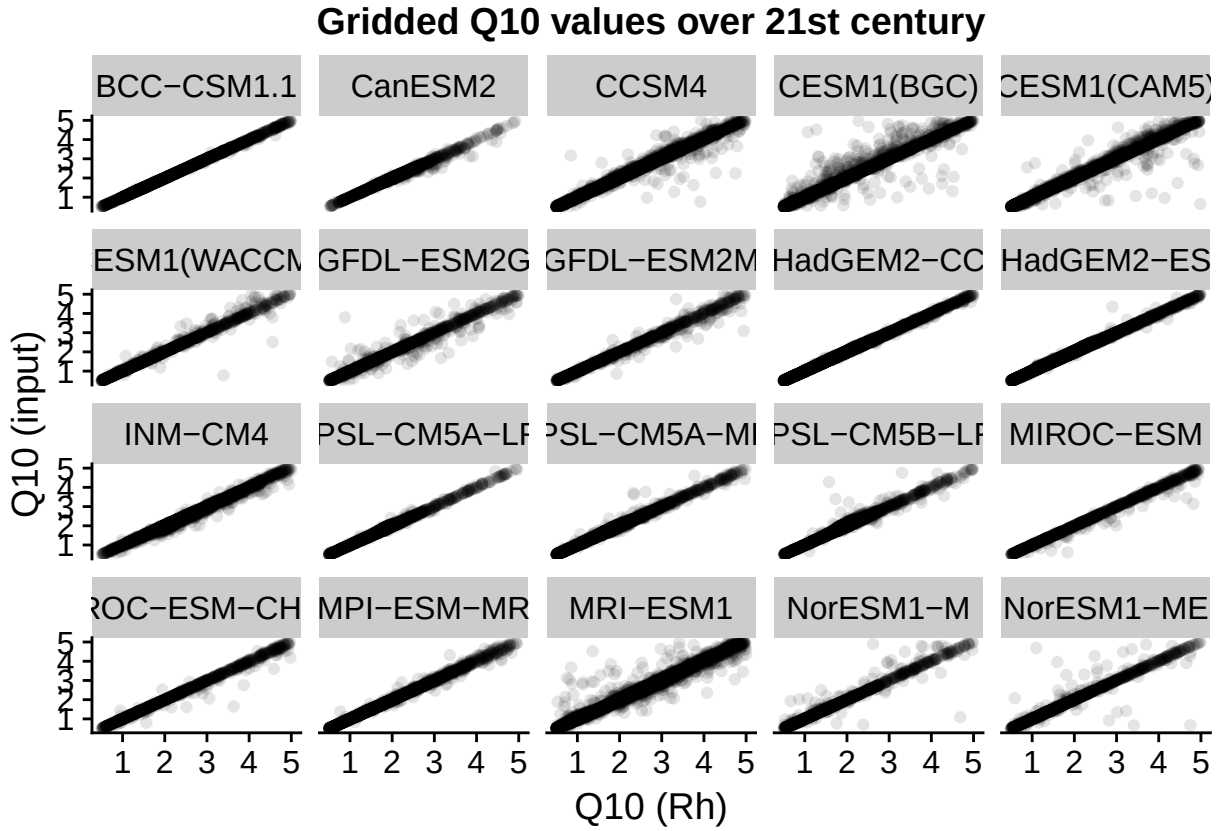


Figure S6: A grid by grid comparison across models of Q10 calculated with the metastable assumption (input) and without the metastable assumption (Rh).

Global mean Q10

```
Q10.mean <- dC.maps %>%
  #filter(is.finite(Q10.dRh), Q10.dRh > 0.5, Q10.dRh < 5) %>%
  filter(is.finite(Q10.dIn), Q10.dIn > 0.5, Q10.dIn < 5) %>%
  #filter(startYrStr == '2006-2015', endYrStr == '2090-2099') %>%
  #select(model, startYrStr, endYrStr, gapYrs, Q10.dRh) %>%
  #summarize(Q10.mean = round(mean(Q10.dRh), 1))
  select(model, model.name, land.model, temperature.model,
         startYrStr, endYrStr, gapYrs, Q10.dIn) %>%
```

```
  group_by(model, model.name, land.model, temperature.model, gapYrs) %>%
  summarize(Q10.mean = round(mean(Q10.dIn), 1),
            count=length(Q10.dIn)) %>%
  filter(gapYrs == 84)

#ggplot(Q10.mean) +
#  geom_line(aes(x=gapYrs, y=Q10.mean, color=model.name)) +
#  facet_wrap(~land.model)

#Q10.mean %>% select(model.name, Q10.mean) %>% print
```

Figure S7: Map of Q10 types and associated SOC change

```
map_plot <- ggplot(data=dC.maps %>%
         filter(startYrStr == '2006-2015', endYrStr=='2090-2099') %>%
         mutate(Q10 = Q10.dIn) %>%
         mutate(Q10Status = if_else(!is.finite(Q10), 'nonfinite',
                                    if_else(Q10 < 0.5, 'Q10 < 0.5',
                                            if_else(Q10 > 5, 'Q10 > 5',
                                                    'typical'))))
         ) +
  geom_raster(aes(x=if_else(lon >0, lon, lon+360), y=lat, fill=Q10Status)) +
  coord_equal() +
  scale_x_continuous(expand=c(0,0)) +
  scale_y_continuous(expand=c(0,0), limits=c(-60, 85)) +
  theme_bw() +
  labs(title='Map of Q10 types') +
  theme(text=element_text(size=14, face='bold'),
        axis.title=element_blank(),
        axis.text=element_blank(),
        axis.ticks=element_blank()) +
  guides(fill=FALSE) +
  facet_wrap(~model.name)

soc_plot <- ggplot(dC.maps %>%
         filter(startYrStr == '2006-2015', endYrStr=='2090-2099') %>%
         mutate(Q10 = Q10.dIn) %>%
         mutate(Q10Status = if_else(!is.finite(Q10), 'nonfinite',
                                    if_else(Q10 < 0.5, 'Q10 < 0.5',
                                            if_else(Q10 > 5, 'Q10 > 5',
                                                    'typical')))) %>%
         group_by(model, startYrStr, endYrStr, Q10Status) %>%
         summarise(org_dSOC = sum((SOC.end-SOC.start)*landArea/1e12, na.rm=TRUE))) +
  geom_col(aes(x=model, y=org_dSOC, fill=Q10Status)) +
  labs(y='Change in 21st century SOC [Pg-C]',
       title='Change in SOC by type of Q10',
       fill='Q10 type') +
  theme_bw() +
  theme(text=element_text(size=14, face='bold'),
        axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),
        axis.title.x = element_blank())
```

```
plot_grid(map_plot, soc_plot, labels = c("A", "B"), nrow=2)
```

```
## Warning: Removed 39402 rows containing missing values (geom_raster).
```

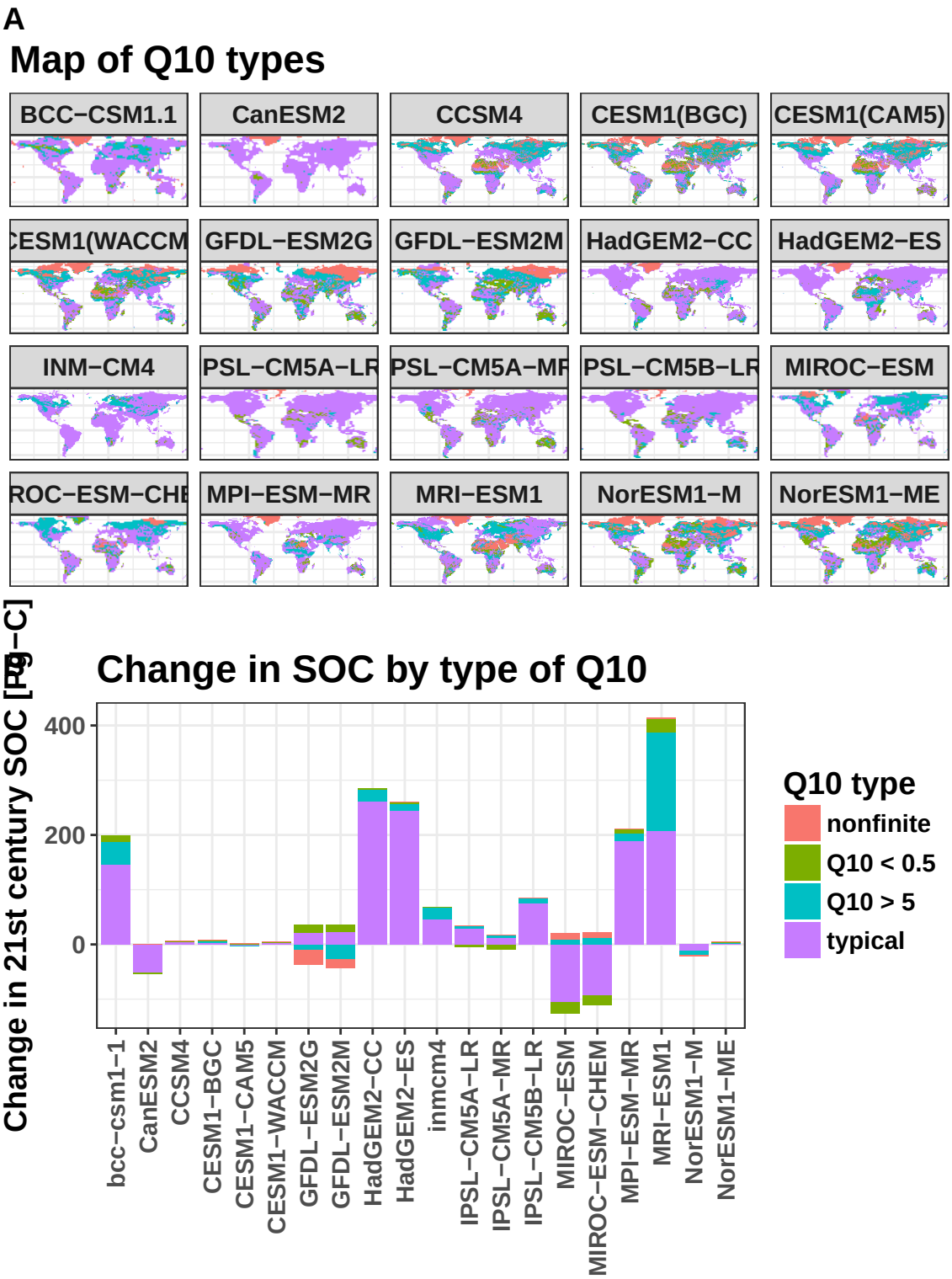


Figure S7: Q10 values calculated over the 21st century can be broken into four categories: non-finite, values less then 0.5, values greater then 5, and values that fall within a typical range of 0.5 to 5. A) Shows the

greographic pattern of these categories and B) the change in soil carbon stock for each category over the 21st century.

Figure 3: Map of 21st centry gridded Q10 values

```
Q10maps.pl <- ggplot(data=dC.maps %>%
         filter(startYrStr == '2006-2015', endYrStr=='2090-2099') %>%
         mutate(Q10 = Q10.dIn) %>%
         mutate(Q10Status = if_else(!is.finite(Q10), 'nonfinite',
                                    if_else(Q10 < 0.5, 'Q10 < 0.5',
                                            if_else(Q10 > 5, 'Q10 > 5',
                                                    'typical')))) %>%
         mutate(Q10 = if_else(grepl('typical', Q10Status), Q10.dIn, as.numeric(NA))))+
  geom_raster(aes(x=if_else(lon >0, lon, lon+360), y=lat, fill=Q10)) +
  scale_fill_gradientn(breaks=1:5, labels=c(as.character(1:4), '5+'),
                       colors=c('blue', 'plum2', 'red'),
                       values=c(0, fieldQ10s$`50%`,  5)/5)+
  coord_equal() +
  scale_x_continuous(expand=c(0,0)) +
  scale_y_continuous(expand=c(0,0), limits=c(-60, 85)) +
  labs(fill=expression(paste('Q'[10]))) +
  theme_bw() +
  theme(axis.title=element_blank(),
        axis.text=element_blank(),
        axis.ticks=element_blank()) +
  facet_wrap(~model.name)

ggsave(Q10maps.pl, filename='temp/Fig3_Q10maps.pdf', width=8, height=6)
```

```
## Warning: Removed 39402 rows containing missing values (geom_raster).
```

```
print(Q10maps.pl)
```

```
## Warning: Removed 39402 rows containing missing values (geom_raster).
```

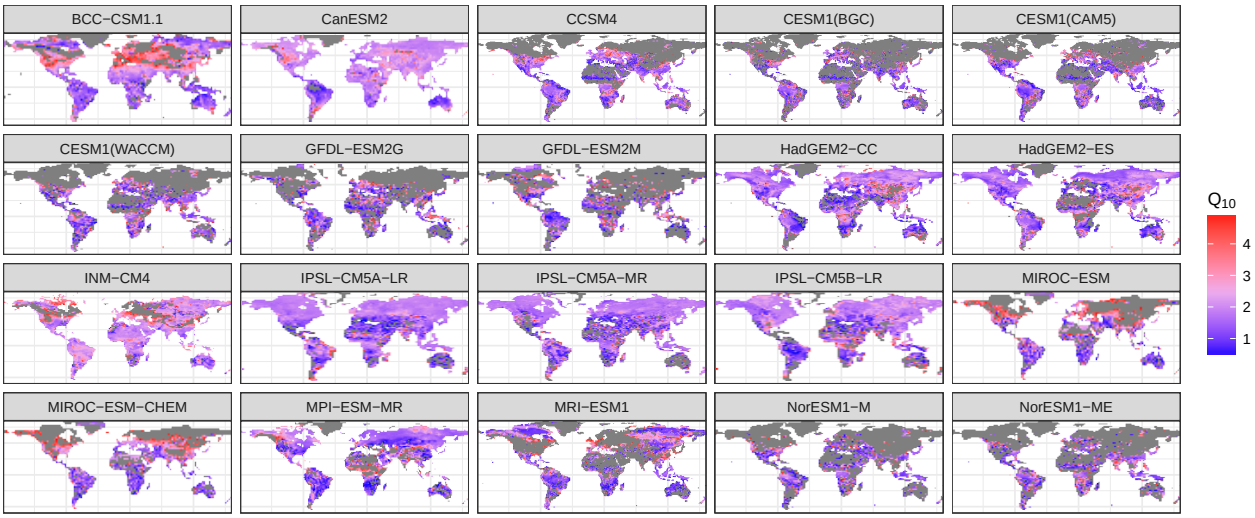


Figure 3: Q10 map for values that fall wihtin the typical range of 0.5 to 5 across model types. Grey indicates Q10 values that are either non-finite, above 5, or below 0.5.

Scale ESM SOC by field Q10 range

```
dC.mapsPlus <- dC.maps %>%
  filter(startYrStr == '2006-2015', endYrStr == '2090-2099') %>%
  left_join(Q10.mean, by=c('model', 'model.name')) %>%
  mutate(Q10=Q10.dIn) %>%
  mutate(SOC.end_05 = if_else(is.finite(Q10) & Q10 > 0.5 & Q10 < 5,
                              SOC.start * input.end / input.start  *
                                (Q10/Q10.mean*fieldQ10s$`5%`) ^
                                ((tsl.start-tsl.end)/10), SOC.end),
         SOC.end_50 = if_else(is.finite(Q10) & Q10 > 0.5 & Q10 < 5,
                              SOC.start * input.end / input.start  *
                                (Q10/Q10.mean*fieldQ10s$`50%`) ^
                                ((tsl.start-tsl.end)/10), SOC.end),
         SOC.end_95 = if_else(is.finite(Q10) & Q10 > 0.5 & Q10 < 5,
                              SOC.start * input.end / input.start  *
                                (Q10/Q10.mean*fieldQ10s$`95%`) ^
                                ((tsl.start-tsl.end)/10), SOC.end))
```

Figure S8 Q10 shift

```
ggplot(dC.mapsPlus %>%
         mutate(Q10=Q10.dIn) %>%
         #filter(is.finite(Q10.dRh), Q10.dRh > 0.5, Q10.dRh < 5) %>%
         filter(is.finite(Q10.dIn), Q10.dIn > 0.5, Q10.dIn < 5) %>%
         filter(startYrStr == '2006-2015', endYrStr == '2090-2099')) +
  geom_density(aes(x=Q10, group=model), fill='grey', color='grey') +
  geom_density(aes(x=Q10/Q10.mean*fieldQ10s$`5%`, group=model), color='blue') +
  geom_density(aes(x=Q10/Q10.mean*fieldQ10s$`50%`, group=model), color='purple') +
  geom_density(aes(x=Q10/Q10.mean*fieldQ10s$`95%`, group=model), color='red') +
  scale_x_log10(breaks=c(0.5, 1, 2, 5)) +
  labs(title='Shifts in gridded Q10', x='Q10 (Input)') +
  theme(axis.title.y=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank()) +
  facet_wrap(~model.name, scales='free_y')
```

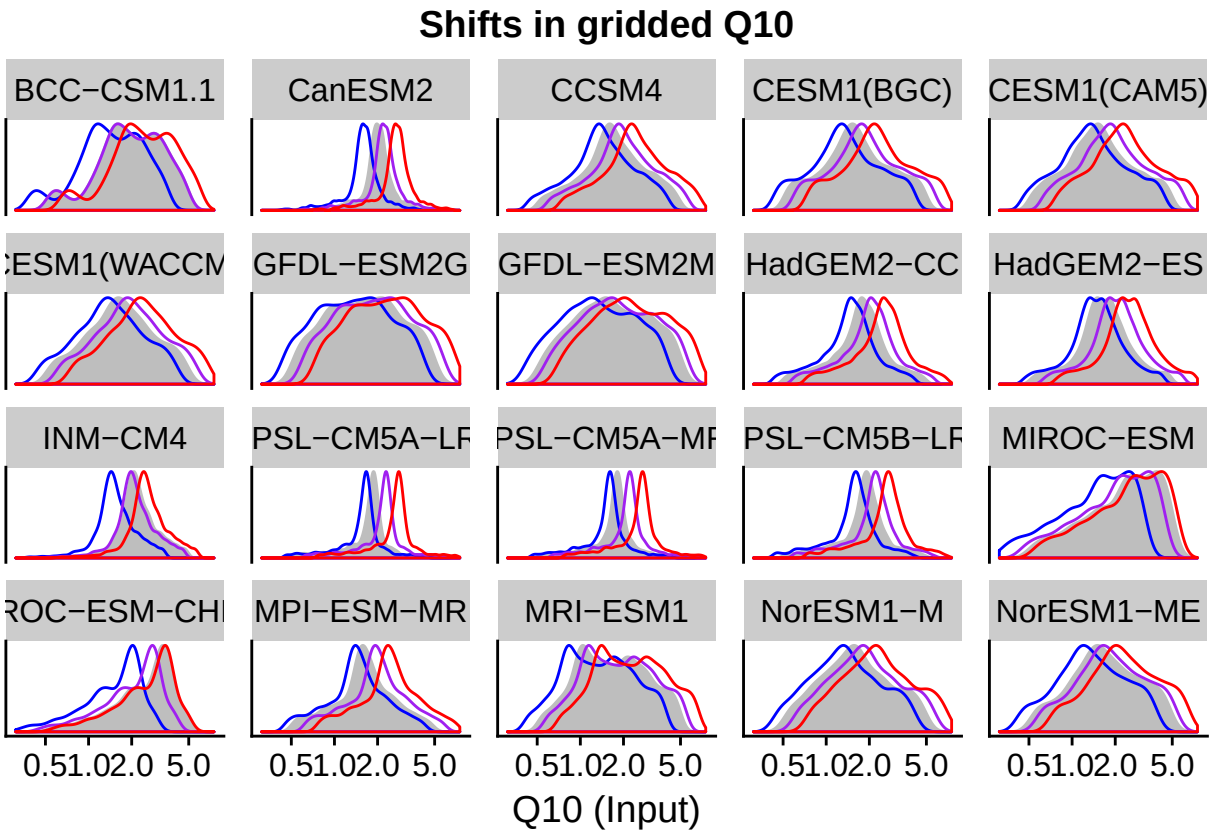


Figure S8: Typical Q10 values where rescaled by multiplying the Q10 to the ratio of the field derived Q10 to the mean Q10. This figure shows the resulting shift in the Q10 distribution for each model. The orginal Q10 distribution is in grey. The shifted distributions are in blue (Q10=1.6), purple (Q10=2.2), and red (Q10=2.7). Note the log-scale on the x-axis.

Table 2: Global totals and summary statistic.

```
dSOC_summary <- dC.mapsPlus %>%
  ungroup() %>% group_by(center, model.name) %>%
  summarise(dSOC_Q10org = sum((SOC.end-SOC.start)*landArea/1e12, na.rm=TRUE),
            dSOC_Q10min= sum((SOC.end_05-SOC.start)*landArea/1e12, na.rm=TRUE),
            dSOC_Q10mid= sum((SOC.end_50-SOC.start)*landArea/1e12, na.rm=TRUE),
            dSOC_Q10max= sum((SOC.end_95-SOC.start)*landArea/1e12, na.rm=TRUE),
            SOC.start=sum(SOC.start*landArea, na.rm=TRUE)/1e12,
            SOC.end=sum(SOC.end*landArea, na.rm=TRUE)/1e12,
            inputRatio = sum(input.end*landArea, na.rm=TRUE)/
              sum(input.start*landArea, na.rm=TRUE),
            dT = sum(dT*landArea, na.rm=TRUE)/sum(is.finite(dT)*landArea, na.rm=TRUE),
            Q10 = unique(Q10.mean))

dSOC_summary <- dSOC_summary %>%
  bind_rows(dSOC_summary %>% ungroup() %>% #calculate the multi-center means
              group_by(center) %>% select(-model.name) %>% summarize_all(mean) %>%
              #by finding within center average
              ungroup() %>% select(-center) %>% summarize_all(mean) %>%
              #Then calcualting average across centers
```

```
            mutate(center = '', model.name = 'Multi-center mean')) %>%
            # rename(SOC.start = SOC.start_mean,
            #        SOC.end =SOC.end_mean,
            #        inputRatio=inputRatio_mean ,
            #        dT=dT_mean ,
            #        Q10=Q10_mean ,
            #        dSOC_Q10org =dSOC_Q10org_mean ,
            #        dSOC_Q10min = dSOC_Q10min_mean,
            #        dSOC_Q10mid= dSOC_Q10mid_mean,
            #        dSOC_Q10max =  dSOC_Q10max_mean)

  bind_rows(dSOC_summary %>% ungroup() %>% #calculate the multi-center means
            group_by(center) %>% select(-model.name) %>% summarize_all(mean) %>%
            #by finding within center average
            ungroup() %>% select(-center) %>% summarize_all(sd) %>%
            #Then calcualting average across centers
            mutate(center = '', model.name = 'Multi-center sd')) %>%
  select(center, model.name, SOC.start, SOC.end, inputRatio, dT, starts_with('Q10'),
         contains('Q10org'), contains('Q10min'), contains('Q10mid'), contains('Q10max'))


write.csv(dSOC_summary, 'temp/Table2.csv')
pander(dSOC_summary, digits=3)
```

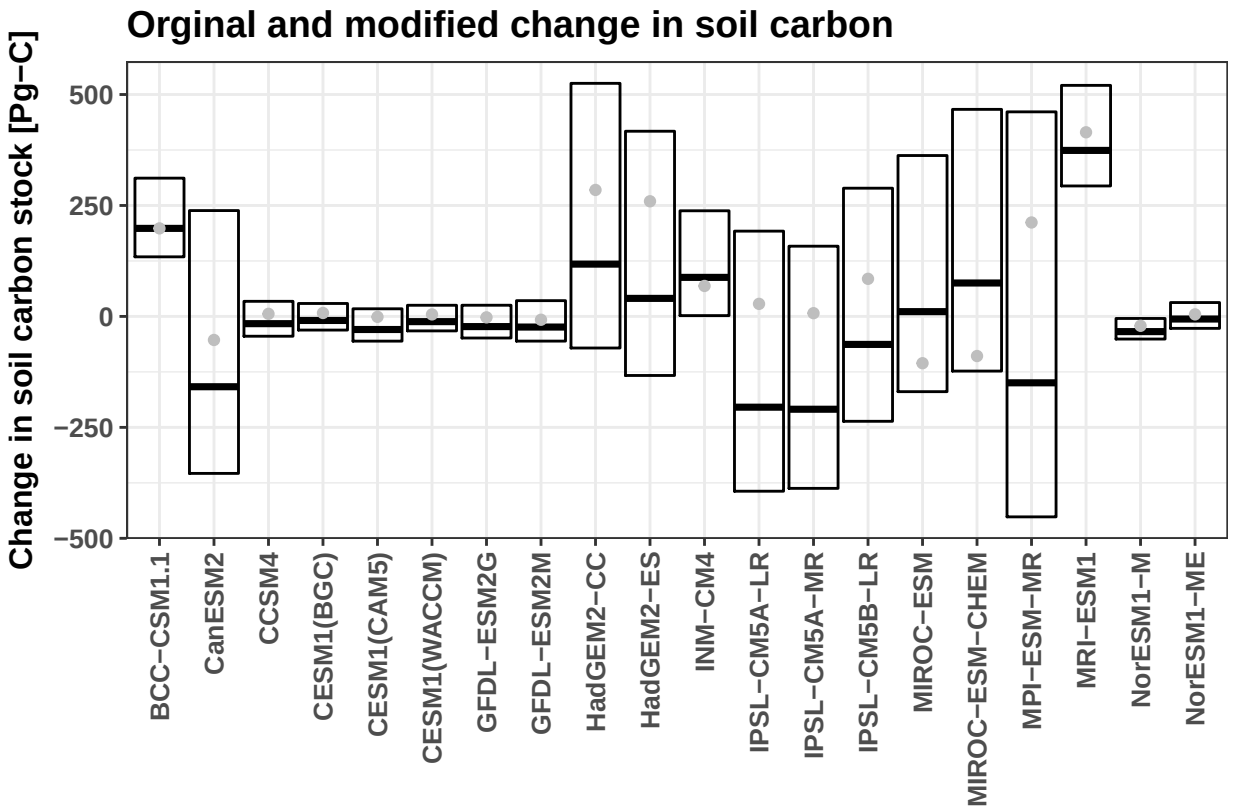
Table 6: Table continues below

center	model.name	SOC.start	SOC.end	inputRatio	dT
BCC	BCC-CSM1.1	1050	1248	1.4	3.7
CCCma	CanESM2	1541	1487	1.29	7.13
INM	INM-CM4	1688	1757	1.27	3.33
IPSL	IPSL-CM5A-LR	1361	1389	1.48	8.18
IPSL	IPSL-CM5A-MR	1403	1410	1.43	7.58
IPSL	IPSL-CM5B-LR	1274	1358	1.41	7.64
MIROC	MIROC-ESM	2586	2481	1.35	7.19
MIROC	MIROC-ESM-CHEM	2588	2498	1.3	7.27
MOHC*	HadGEM2-CC	1122	1407	1.55	8.37
MOHC*	HadGEM2-ES	1129	1389	1.56	8.26
MPI-M	MPI-ESM-MR	3110	3321	1.31	6.33
MRI	MRI-ESM1	1452	1867	1.52	4.36
NCAR	CCSM4	515	521	1.32	4.16
NCC	NorESM1-M	547	526	1.31	3.68
NCC	NorESM1-ME	553	557	1.32	3.59
NOAA GFDL	GFDL-ESM2G	1422	1420	1.41	5.08
NOAA GFDL	GFDL-ESM2M	1278	1270	1.38	4.53
NSF-DOE-NCAR	CESM1(BGC)	515	523	1.29	3.8
NSF-DOE-NCAR	CESM1(CAM5)	553	552	1.3	4.64
NSF-DOE-NCAR	CESM1(WACCM)	502	506	1.32	3.92
	Multi-center mean	1403	1491	1.37	5.41
	Multi-center sd	793	822	0.0924	1.82

Q10	dSOC_Q10org	dSOC_Q10min	dSOC_Q10mid	dSOC_Q10max
2.2	198	312	198	134
2	-53	239	-158	-354
2.3	68.7	238	88	1.82
1.8	28.3	192	-205	-394
1.8	7.29	158	-209	-387
1.9	84.7	289	-63.1	-236
2.5	-105	363	10.8	-170
2.6	-89.2	467	75.3	-123
1.9	285	525	118	-71.2
1.8	259	417	40.6	-133
1.8	212	461	-150	-452
2	415	521	374	294
1.9	5.81	34	-16.3	-44.6
1.9	-21.2	-4.64	-34.2	-51
2	4.63	31.2	-5.89	-26.9
1.9	-2.12	25.2	-22.9	-48.6
2	-7.54	35.6	-23.9	-55.6
1.9	7.76	29.1	-9.11	-30.7
1.8	-1	17.2	-29.5	-55.7
1.9	4.52	25.1	-11.7	-32.6
2.02	87.7	248	19.9	-94.9
0.224	153	191	155	209

Figure 4: Change in ESM SOC estimates with modified Q10

```
dSOC.pl <- ggplot(data = dSOC_summary %>% filter(center != '')) +
  geom_crossbar(aes(x=model.name, y=dSOC_Q10mid, ymin=dSOC_Q10min, ymax=dSOC_Q10max)) +
  geom_point(aes(x=model.name, y=dSOC_Q10org), color='grey') +
  labs(x='', y='Change in soil carbon stock [Pg-C]',
       title='Orginal and modified change in soil carbon') +
  theme_bw() +
  theme(text=element_text(size=12, face='bold'),
        axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

ggsave(dSOC.pl, filename='temp/Fig4_dSOC.pdf', width=6.25, height=4.25)
print(dSOC.pl)
```



Graphical Abstract: Field to ESM temperater sensitivity

```
fieldPlot <- ggplot(data.frame(temperature = -10:35) %>%
  mutate(minQ10 = fieldQ10s$`5%`^((temperature-15)/10),
         midQ10 = fieldQ10s$`50%`^((temperature-15)/10),
         maxQ10 = fieldQ10s$`95%`^((temperature-15)/10)) %>%
  gather(Q10Type, modifier, ends_with('Q10')) %>%
  mutate(Q10 = if_else(Q10Type == 'minQ10', fieldQ10s$`5%`,
                       if_else(Q10Type == 'midQ10', fieldQ10s$`50%`,
                               if_else(Q10Type == 'maxQ10', fieldQ10s$`95%`,
                                       as.numeric(NA))))) +
  geom_line(aes(x=temperature, y=modifier, color=as.factor(Q10)), size=3) +
  labs(x='Temperature [C]',
       y='Decay rate',
       title='Field temperature sensitivity',
       color=expression(Q[10])) +
  scale_y_continuous(breaks=seq(0, 6, by=2),
                     labels=sprintf('%d x Kref', seq(0, 6, by=2))) +
  scale_color_manual(values = c('blue', 'purple', 'red')) +
  theme_bw() +
  theme(text=element_text(size=12), legend.position=c(0.2,0.8))


ESMplot <- ggplot( dSOC_summary %>% ungroup() %>%
                     filter(grepl('Multi', model.name)) %>%
                      mutate(model.name = gsub('Multi-center ', '', model.name)) %>%
```

```
                    select(-center) %>% group_by(model.name) %>%
                    gather(key='variable', value = 'value', -model.name) %>%
                    spread(model.name, value) %>%
                    filter(grepl('dSOC', variable)) %>%
                    separate(variable, into=c('type', 'Q10')) %>%
                    mutate_at(vars(Q10), factor)) +
  geom_crossbar(aes(x=Q10, y=mean, ymin=mean-sd,
                    ymax=mean+sd, color=Q10), size=3) +
  scale_color_manual(values = c(Q10org='black', Q10min='blue',
                                Q10mid='purple', Q10max='red'), guide=FALSE) +
  scale_x_discrete(breaks=c('Q10org', 'Q10min', 'Q10mid', 'Q10max'),
                   labels = c('Orginal', fieldQ10s$`5%`,
                              fieldQ10s$`50%`, fieldQ10s$`95%`)) +
  labs(y='Change in soil carbon stock [Pg]',
       x = expression(paste('Scaled Q'[10],' mode')),
       title=expression(paste('Soil carbon loss with modified Q'[10])))+
  theme_bw() +
  theme(text=element_text(size=12))

GAPlot <- plot_grid(fieldPlot, ESMplot, labels = c("A", "B"))
ggsave('temp/graphicalAbstract.pdf', GAPlot, width=8, height=4)
print(GAPlot)
```