LMM_NvS

Created January 28, 2025

Changes

- 1/28/25: loading data

```r
library(lme4)
```

```
## Loading required package: Matrix
```

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)
```

Load Data

```r
# File from github
filepath = "https://raw.githubusercontent.com/shabanm2/Utqiagvik/pca-25/Analysis_Ready_Data/"
df <- read.csv(paste0(filepath, "daily_2022_2024.csv"))
df <- df %>% select(-X) %>% select(-X.1)
df$Date <- as.POSIXct(df$date, format="%Y-%m-%d")
df <- df %>% filter(windspeed != -888.88) %>% filter(winddir != -888.88)
```

Select and Transform Data

North vs South

TNHA:

North = TNHA-SC

South = TNHA-SA

SSMH:

North = SSMH-SB

South = SSMH-SA

BEO (Control): does not have different aspects

```
nvs <- df %>% filter(fullname == "TNHA-SA" | fullname == "TNHA-SC" | fullname == "SSMH-SB" | fullname =
```

```
# filter out data from before data collection
# filter to get only depth of 10cm for now
df_10cm <- nvs %>% filter(grounddepth == 8) %>% filter(Date >= "2022-06-19") %>% filter(Date < "2022-09
```

Fit LMM

```
lmm0 <- lmer(groundtemp ~ airtemp + vwc + solar + windspeed + aspect + (1|site), data = df_10cm)
summary(lmm0)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: groundtemp ~ airtemp + vwc + solar + windspeed + aspect + (1 |
##     site)
##    Data: df_10cm
##
## REML criterion at convergence: 955.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.1222 -0.6453 -0.0094  0.5647  2.5418
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  site     (Intercept) 0.3026   0.5501
##  Residual             1.3784   1.1741
## Number of obs: 296, groups:  site, 2
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  3.063267   0.524690   5.838
## airtemp      0.504097   0.028961  17.406
## vwc         -3.909724   0.877143  -4.457
## solar        0.008013   0.001483   5.402
## windspeed   -0.461289   0.123771  -3.727
## aspectSouth  0.412346   0.152185   2.710
##
## Correlation of Fixed Effects:
##             (Intr) airtmp vwc    solar  wndspd
## airtemp     -0.169
## vwc         -0.542  0.021
## solar       -0.165 -0.464  0.112
```

```
## windspeed   -0.375  0.000  0.247  0.082
## aspectSouth -0.167  0.051  0.058 -0.288  0.293
```

It seems like there is not too much of a difference between the two sites, but there is still presence of a difference as shown by the two different intercepts for SSMH vs TNHA.

Graphing linear models: Random effects have different intercepts, and the fixed effects have different slopes

```
coeffs = coef(lmm0)$site
coeffs
```

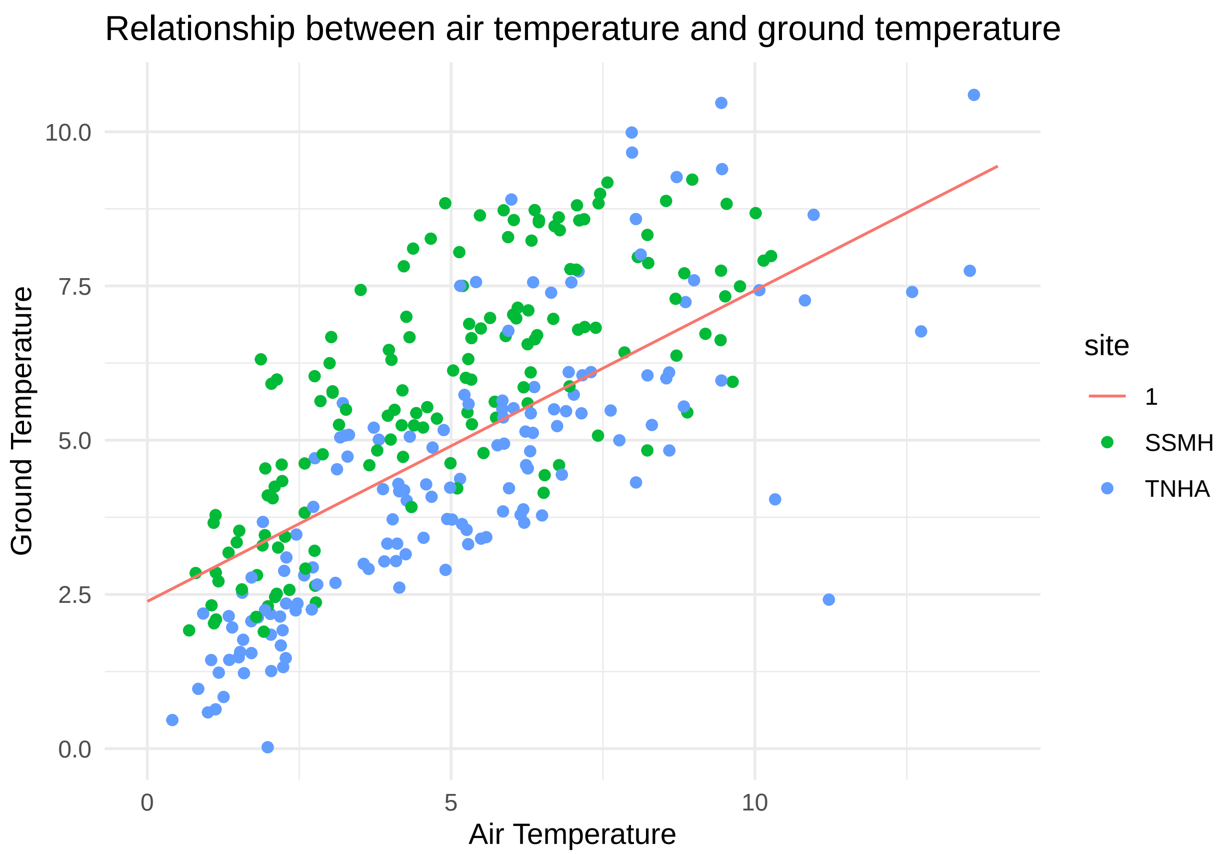
```
##      (Intercept)   airtemp       vwc       solar  windspeed aspectSouth
## SSMH    3.443036 0.5040974 -3.909724 0.008012815 -0.4612893   0.4123458
## TNHA    2.683499 0.5040974 -3.909724 0.008012815 -0.4612893   0.4123458
```

```
library(ggeffects)  # install the package first if you haven't already, then load it

# Extract the prediction data frame
pred.mm <- ggpredict(lmm0, terms = c("airtemp"))  # this gives overall predictions for the model

# Plot the predictions

(ggplot(pred.mm) +
    geom_point(data = df_10cm,                      # adding the raw data (scaled values)
              aes(x = airtemp, y = groundtemp, colour = site)) +
   geom_line(aes(x = x, y = predicted, color = group)) +          # slope
   #geom_ribbon(aes(x = x, ymin = predicted - std.error, ymax = predicted + std.error),
               #fill = "lightgrey", alpha = 0.5) +  # error band
   labs(x = "Air Temperature", y = "Ground Temperature",
        title = "Relationship between air temperature and ground temperature") +
   theme_minimal()
)
```




```
# i don't like this one
#library(ggeffects)
#ggpredict(lmm1, terms = c("airtemp","aspect", "site"), type = "random") %>%
#   plot() +
#   labs(x = "Body Length", y = "Test Score", title = "Effect of body size on intelligence in dragons")
#   theme_minimal()
```

Table Output

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
stargazer(lmm0, type = "text",
          digits = 3,
          star.cutoffs = c(0.05, 0.01, 0.001),
          digit.separator = "")
```

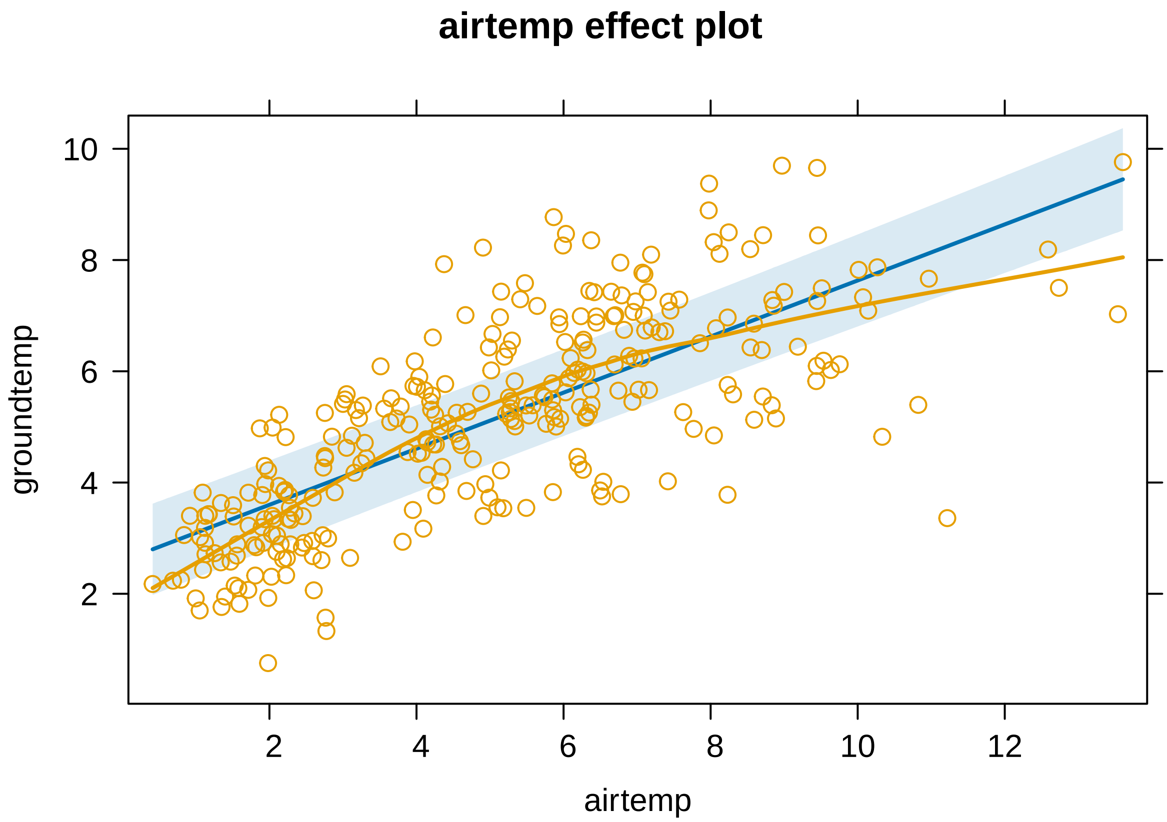
```
##
## ===================================================
##                          Dependent variable:
##                      ----------------------------
##                              groundtemp
## ---------------------------------------------------
## airtemp                       0.504***
##                                (0.029)
##
## vwc                           -3.910***
##                                (0.877)
##
## solar                         0.008***
##                                (0.001)
##
## windspeed                     -0.461***
##                                (0.124)
##
## aspectSouth                    0.412**
##                                (0.152)
##
## Constant                      3.063***
##                                (0.525)
##
## ---------------------------------------------------
## Observations                     296
## Log Likelihood                -477.861
## Akaike Inf. Crit.              971.722
## Bayesian Inf. Crit.           1001.245
## ===================================================
## Note:                 *p<0.05; **p<0.01; ***p<0.001
```

```r
library(effects)
```

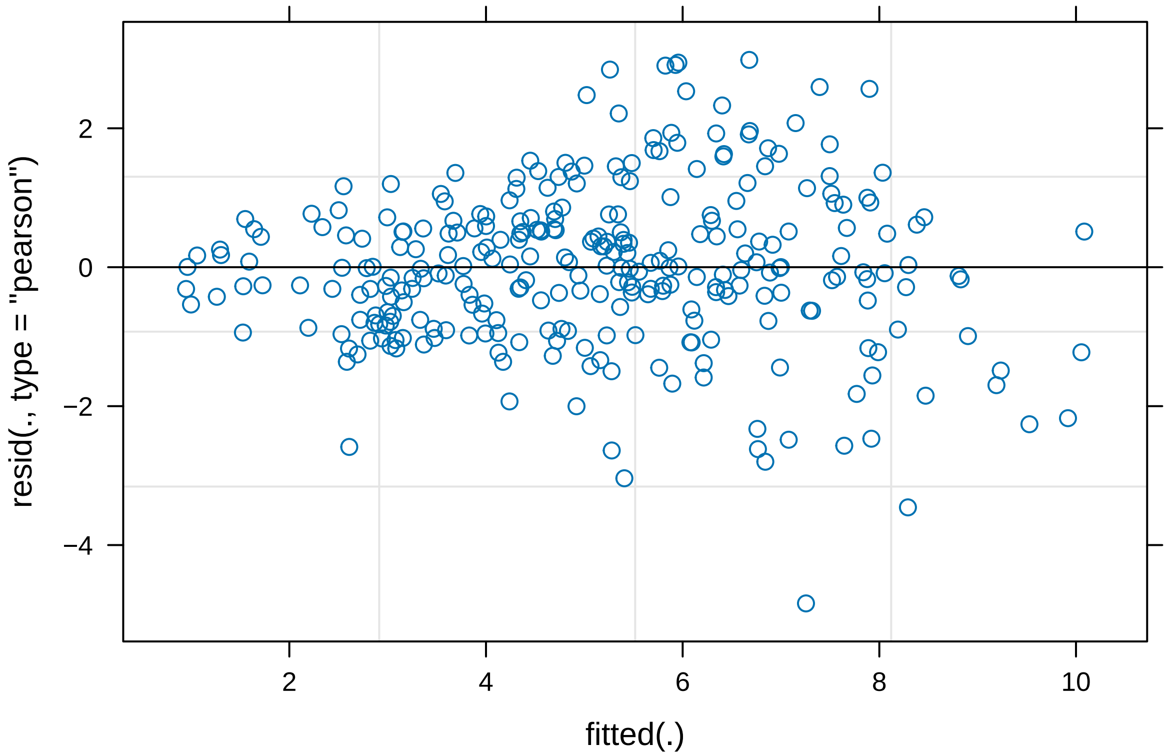
```
## Loading required package: carData
```

```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```r
est<-Effect("airtemp", partial.residuals=T, lmm0)
plot(est)
```

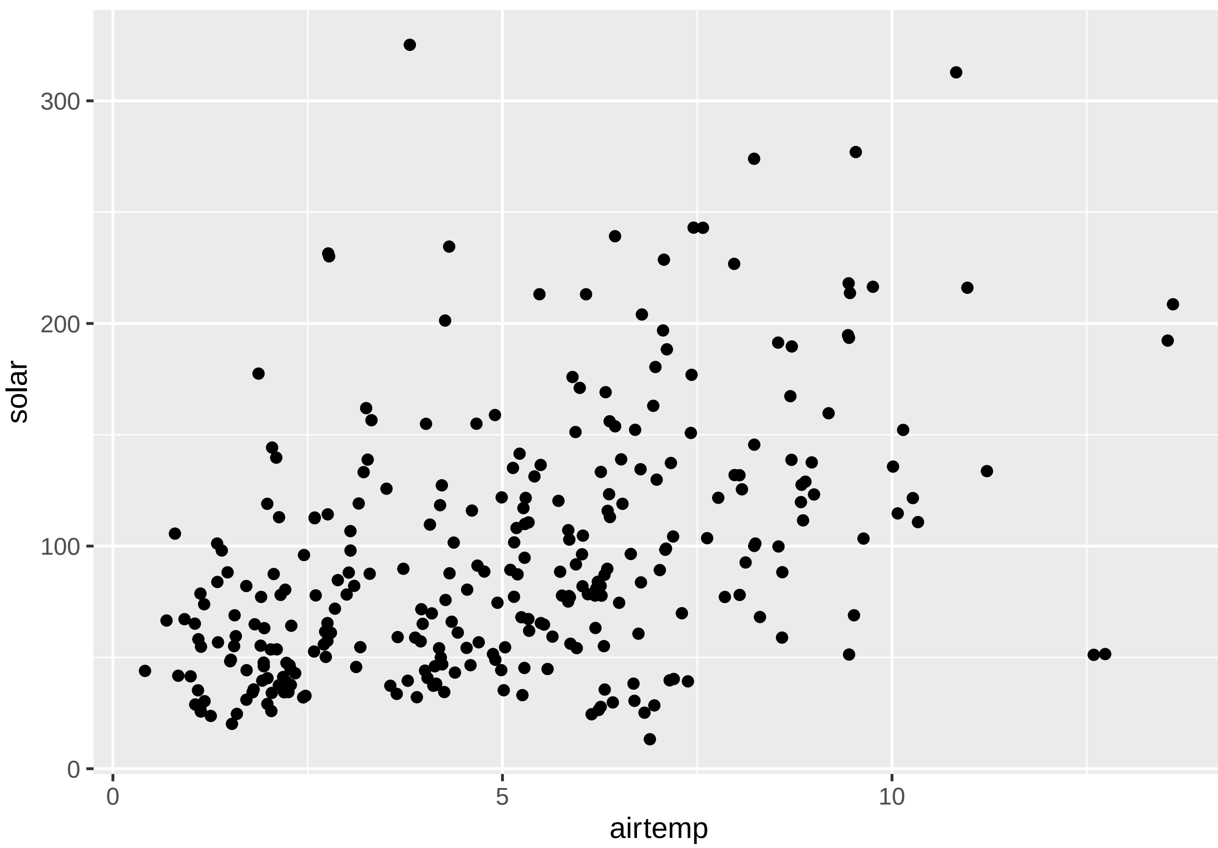



```
plot(lmm0)
```




```
#library(remef)
#y_partial <- remef(model, fix = "x2", ran = "all")
```

```
ggplot(df_10cm, aes(x=airtemp, y=solar)) + geom_point()
```


solar
airtemp
0
100
200
300
0
5
10