

Modelling Maize Project: height_2022_07_05.num

Modelling with smooth effect

Author: Nisia Trisconi | Zurich Data Scientists
Reviewer: Dr. Luisa Barbanti | Zurich Data Scientists

October 30, 2023

Contents

1	Freezing Package versions	2
2	Load packages	2
3	Settings	2
4	Getting data	3
5	Design	5
6	Response variable: <i>height_2022_07_05.num</i>	5
6.1	Aim	5
6.2	Model fitting	5
6.3	Model selection	7
6.3.1	Logarithmic transformation	7
6.3.2	Smooth terms	11
6.4	Model checking	16
7	Methods description	17
8	Session information	19

1 Freezing Package versions

The following code lines are commented out because the `{checkpoint}` package no longer works.

```
## (messages are omitted in this chunk)
##
# library(checkpoint)
# checkpoint(snapshot_date = "2022-11-15")
```

2 Load packages

```
## (messages are omitted from this chunk)
##
library(dplyr)
library(kableExtra)
library(ggplot2)
library(tibble) ## function rownames_to_column()
library(multcomp)
library(mgcv)
```

3 Settings

Global settings:

```
Sys.setenv(lang = "en_US")
theme_set(theme_bw())

if (!dir.exists("Prepared_data_and_models")) {
  dir.create("Prepared_data_and_models")
}
```

4 Getting data

```
d.maize <- readRDS(file = paste0("Prepared_data_and_models/",  
                                "d.maize_PreparedData.RDS"))
```

Overview of the data:

```
dim(d.maize)
```

```
[1] 108 33
```

```
head(d.maize)[1:min(ncol(d.maize), 30)]
```

```
# A tibble: 6 x 30  
  pot    soil      well depth seed.weight fungus date.germinated observations  
  <chr> <chr>    <chr> <dbl>    <dbl> <chr>    <chr>          <chr>  
1 A1    Bio garden a      3      30 <NA>    2022-05-11    <NA>  
2 A1    Bio garden b      5      34 <NA>    2022-05-11    <NA>  
3 A1    Bio garden c      2      35 <NA>    2022-05-09    <NA>  
4 A1    Bio garden d      1      40 <NA>    2022-05-10    <NA>  
5 A1    Bio garden e      4      46 <NA>    2022-05-11    <NA>  
6 A1    Bio garden f      6      37 <NA>    2022-05-11    <NA>  
# i 22 more variables: height_2022_07_05 <chr>, cob_weight <chr>, ...12 <dbl>,  
# pot.fac <fct>, soil.fac <fct>, well.fac <fct>, seed.weight.grams <dbl>,  
# fungus.fac <fct>, date.germinated.asDate <date>, obs.time <fct>,  
# broken <lgl>, height_2022_07_05.num <dbl>, plant.found <lgl>,  
# cob_weight.num <dbl>, germinated.in.lab <lgl>, germinated.in.field <lgl>,  
# germinated.yes <lgl>, days.to.germination <dbl>,  
# days.to.germination.censored <dbl>, seed_coord_y <dbl>, ...
```

```
str(d.maize)
```

```
tibble [108 x 33] (S3: tbl_df/tbl/data.frame)  
$ pot           : chr [1:108] "A1" "A1" "A1" "A1" ...  
$ soil          : chr [1:108] "Bio garden" "Bio garden" "Bio garden" "Bio garden" ...  
$ well          : chr [1:108] "a" "b" "c" "d" ...  
$ depth         : num [1:108] 3 5 2 1 4 6 6 4 5 1 ...  
$ seed.weight   : num [1:108] 30 34 35 40 46 37 27 16 23 22 ...  
$ fungus        : chr [1:108] NA NA NA NA ...  
$ date.germinated : chr [1:108] "2022-05-11" "2022-05-11" "2022-05-09" "2022-05-10" ...  
$ observations   : chr [1:108] NA NA NA NA ...  
$ height_2022_07_05 : chr [1:108] "217" "131" "143" "194" ...  
$ cob_weight     : chr [1:108] "117" "26" "61" "109" ...  
$ ...12         : num [1:108] NA NA NA NA NA NA NA NA NA NA ...  
$ pot.fac       : Factor w/ 18 levels "A1","A2","A3",...: 1 1 1 1 1 1 2 2 2 2 ...  
$ soil.fac      : Factor w/ 4 levels "Bio garden","Composana",...: 1 1 1 1 1 1 3 3 3 3 ..  
$ well.fac      : Factor w/ 6 levels "a","b","c","d",...: 1 2 3 4 5 6 1 2 3 4 ...  
$ seed.weight.grams : num [1:108] 0.3 0.34 0.35 0.4 0.46 0.37 0.27 0.16 0.23 0.22 ...  
$ fungus.fac     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...  
$ date.germinated.asDate : Date[1:108], format: "2022-05-11" "2022-05-11" ...  
$ obs.time       : Factor w/ 2 levels "morning","night": 2 2 2 2 2 2 2 2 2 2 ...  
$ broken         : logi [1:108] FALSE FALSE FALSE FALSE FALSE FALSE ...  
$ height_2022_07_05.num : num [1:108] 217 131 143 194 206 233 158 282 241 232 ...  
$ plant.found    : logi [1:108] TRUE TRUE TRUE TRUE TRUE TRUE ...  
$ cob_weight.num  : num [1:108] 117 26 61 109 106 156 57 286 51 120 ...  
$ germinated.in.lab : logi [1:108] TRUE TRUE TRUE TRUE TRUE TRUE ...
```

```

$ germinated.in.field      : logi [1:108] FALSE FALSE FALSE FALSE FALSE FALSE ...
$ germinated.yes          : logi [1:108] TRUE TRUE TRUE TRUE TRUE TRUE ...
$ days.to.germination      : num [1:108] 11 11 9 10 11 11 11 11 NA 9 ...
$ days.to.germination.censored: num [1:108] 11 11 9 10 11 11 11 11 14 9 ...
$ seed_coord_y            : num [1:108] 1 1 2 2 3 3 1 1 2 2 ...
$ seed_coord_x            : num [1:108] 1 2 1 2 1 2 3 4 3 4 ...
$ position_field_x        : num [1:108] 1 1 1 1 1 1 1 1 1 1 ...
$ position_field_x_cm     : num [1:108] 50 50 50 50 50 50 50 50 50 50 ...
$ position_field_y        : int [1:108] 1 2 3 4 5 6 7 8 9 10 ...
$ position_field_y_cm     : num [1:108] 25 50 75 100 125 150 175 200 225 250 ...

```

5 Design

108 maize seeds are planted in 18 different pots, each with 6 wells.

Inside one pot, the same soil is used. The soils that were used are: Bio garden (4 pots), Composana (4 pots), herbs (6 pots), mixture (4 pots).

In each well, one maize seed is planted at a pre-defined depth (in cm), which is allocated randomly to the well. The maximum value for depth is 6cm and this corresponds to planting the seed directly in the coconut fiber that makes up the pot.

Wells in the same pots are allocated as follows:

```
[,1] [,2]
[1,] "e" "f"
[2,] "c" "d"
[3,] "a" "b"
```

The pots are arranged as follows on a table in the lab:

```
[,1] [,2] [,3] [,4] [,5] [,6]
[1,] "C1" "C2" "C3" "C4" "C5" "C6"
[2,] "B1" "B2" "B3" "B4" "B5" "B6"
[3,] "A1" "A2" "A3" "A4" "A5" "A6"
```

Seeds are watered for the first time on 04.30.2022 and are transferred to the field on 05.15.2022 according to the same scheme.

Some seeds are broken when planted, one seed develops a fungus.

Some seeds germinate in the lab, others in the field, while some seeds never germinate.

On 07.05.2022, the height of all maize plants is measured in cm. The plants that were not measured for time reasons receive a height value of `not measured`, while the plants that were not found and could hence not be measured present missing values.

On 09.16.2022, the weight of the cob is measured for all plants that have a cob. The variety of maize that was planted typically yields 1 cob per plant.

6 Response variable: *height_2022_07_05.num*

6.1 Aim

We are interested in testing whether *height_2022_07_05.num* is influenced by the following variables:

- Position in the field (i.e. *position_field_x_cm* and *position_field_y_cm*)
- Soil (variable *soil.fac*)
- Depth in soil (variable *depth*)
- Seed weight (variable *seed.weight*)

6.2 Model fitting

The variable *height_2022_07_05.num* is a continuous variable (in particular it is an amount), whose density is already well-centered, consequently there should be no need to log-transform it.

position_field_x_cm and *position_field_y_cm* represent the position of seeds in the field. They are numeric variables which are introduced in the model as smooth variables, in a bi-dimensional way.

For this reason, a generalised additive model is fitted, using the `gam()` function in the `{mgcv}` package.

```
gam.height_2022_07_05.num <- gam(height_2022_07_05.num ~
  s(position_field_x_cm,
    position_field_y_cm) +
  soil.fac +
  depth +
  seed.weight,
  data = d.maize)
```

```
##
summary(gam.height_2022_07_05.num)
```

Family: gaussian
Link function: identity

Formula:

```
height_2022_07_05.num ~ s(position_field_x_cm, position_field_y_cm) +
  soil.fac + depth + seed.weight
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	188.854219	39.160071	4.8226	1.336e-05	***
soil.facComposana	70.038053	31.407757	2.2300	0.03021	*
soil.facherbs	38.254407	21.762687	1.7578	0.08482	.
soil.facmixture	34.550145	25.206473	1.3707	0.17652	
depth	-0.273933	3.684139	-0.0744	0.94102	
seed.weight	-0.067274	0.984288	-0.0683	0.94578	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

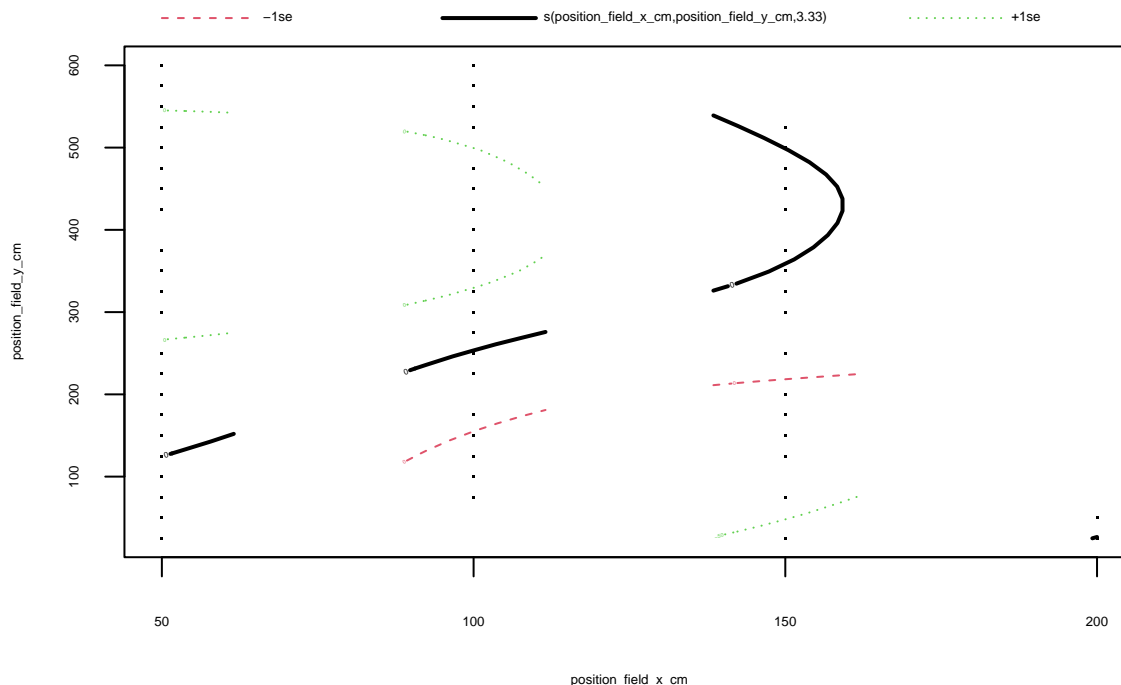
Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(position_field_x_cm,position_field_y_cm)	3.3342	4.3281	1.3013	0.2729

R-sq.(adj) = 0.0863 Deviance explained = 21.5%

GCV = 2576.5 Scale est. = 2175.7 n = 60

```
plot(gam.height_2022_07_05.num)
```



The “Composana” soil shows evidence of being statistically different from the reference level of soil “Bio garden”.

The effective degrees of freedom (referred to as edf) are higher than 2, suggesting that the position in the field has a non-linear relationship with the response variable. The plot displays the gradient of the estimated smooth terms, confirming a non-linear trend.

The explained deviance is low, indicating that the model does not accurately represent the true distribution of the data set.

6.3 Model selection

6.3.1 Logarithmic transformation

As introduced above, the variable *height_2022_07_05.num* is an amount, thus usually it is log transformed inside the model.

Since the graphical analysis did not indicate any need for applying this transformation, we are more inclined to keep the model as simple as possible. However, we fitted the model with the log-transformed response variable to see whether it could lead to an improvement, as the previous model showed evidence of being a poor fit.

```
# The response variable is amount, thus we log transform it and refit the model
gam.height_2022_07_05.num.log <- gam(log(height_2022_07_05.num) ~
  s(position_field_x_cm,
    position_field_y_cm) +
  soil.fac +
  depth +
  seed.weight,
```

```

data = d.maize)

##
summary(gam.height_2022_07_05.num.log)

Family: gaussian
Link function: identity

Formula:
log(height_2022_07_05.num) ~ s(position_field_x_cm, position_field_y_cm) +
  soil.fac + depth + seed.weight

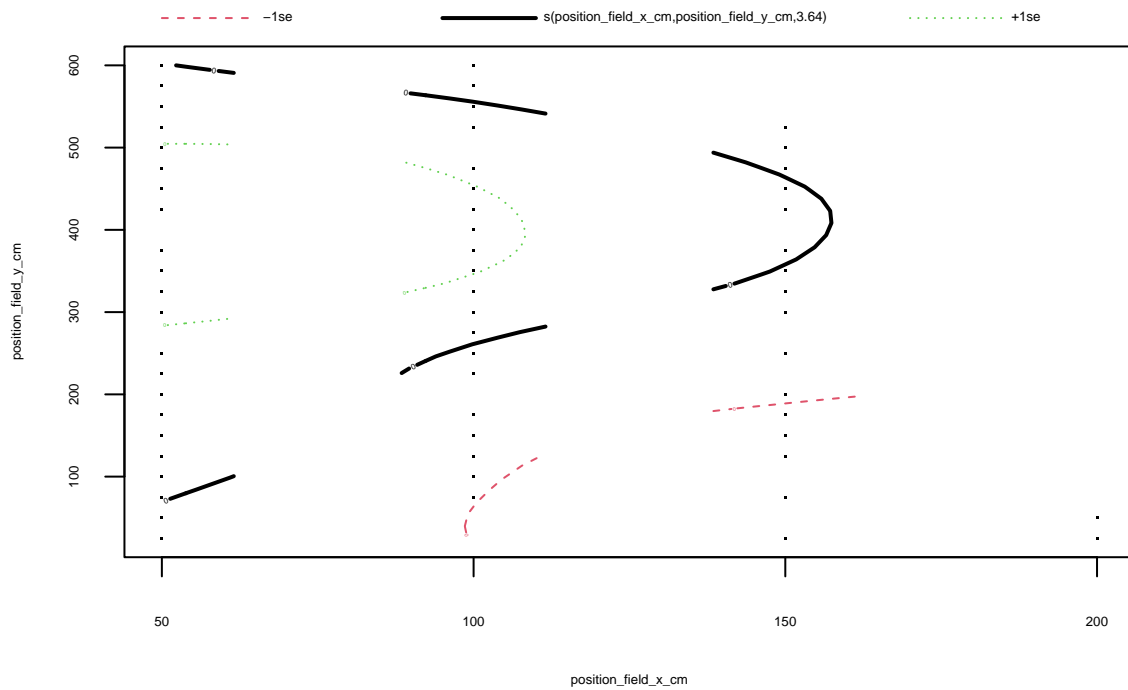
Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    5.2199545   0.2414226  21.6216 < 2e-16 ***
soil.facComposana 0.3951062   0.1954756   2.0213  0.04858 *
soil.facherbs    0.2038920   0.1364329   1.4944  0.14130
soil.facmixture  0.2083776   0.1562637   1.3335  0.18837
depth           -0.0014215   0.0226762  -0.0627  0.95027
seed.weight      -0.0015888   0.0060536  -0.2624  0.79405
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(position_field_x_cm,position_field_y_cm) 3.6434 4.8072 0.8034  0.529

R-sq.(adj) = 0.054  Deviance explained = 19.3%
GCV = 0.097683  Scale est. = 0.081983  n = 60

plot(gam.height_2022_07_05.num.log)

```



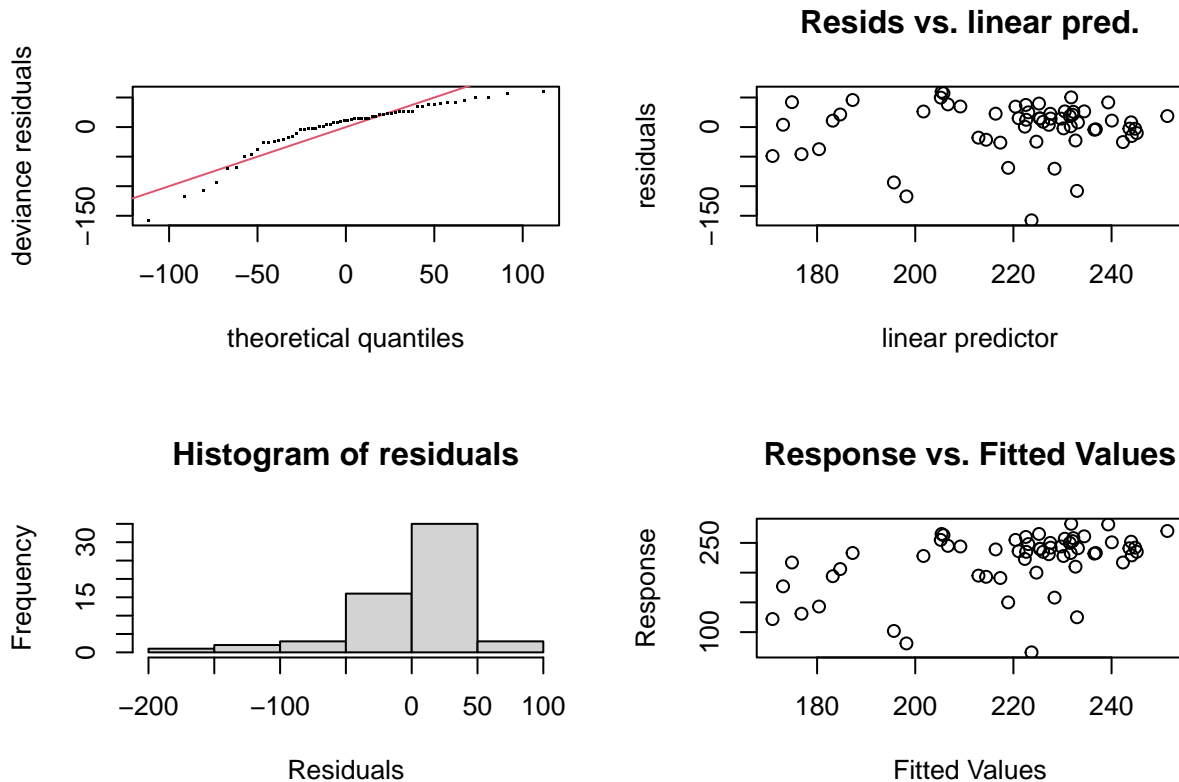
The deviance explained is even worse than before.

The effective degrees of freedom (EDF) and the plot of the model closely resemble those of the previous model.

As previously mentioned the two models are on different scales, thus we cannot compare them using the AIC and BIC criteria.

Our attempt to determine the better model is based on an analysis of the residual plots.

```
gam.check(gam.height_2022_07_05.num)
```



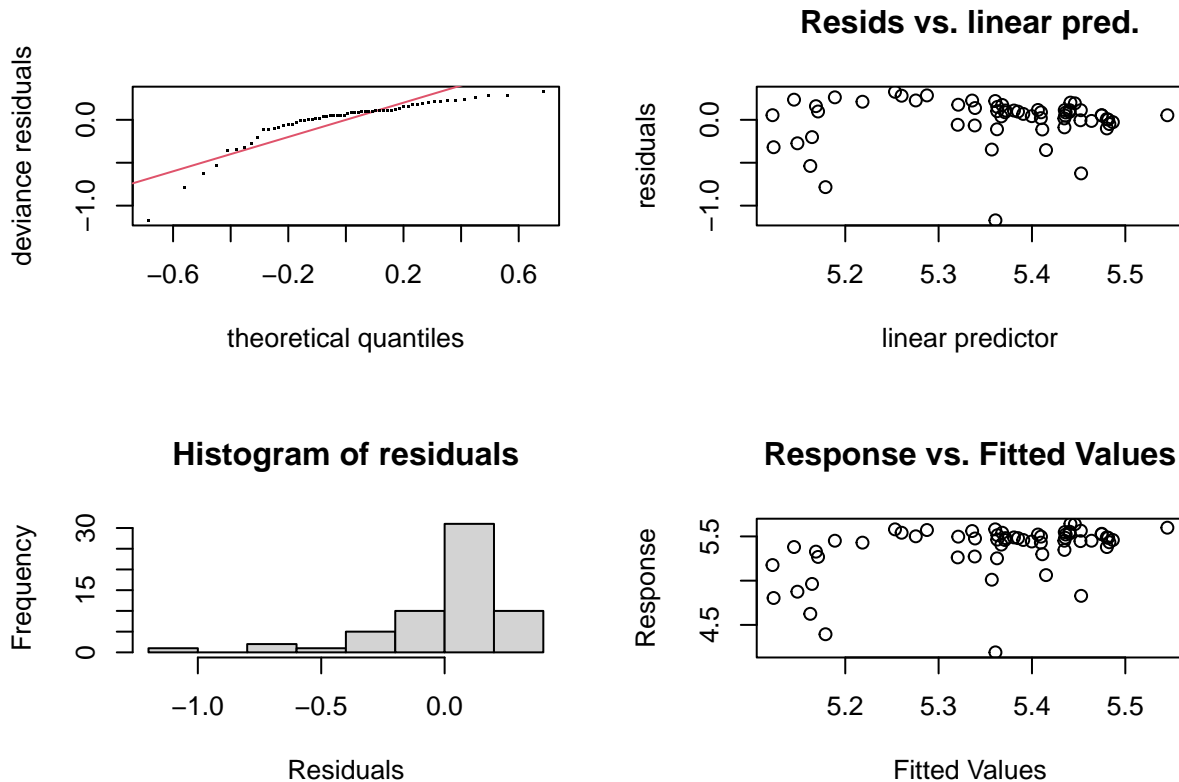
Method: GCV Optimizer: magic
 Smoothing parameter selection converged after 9 iterations.
 The RMS GCV score gradient at convergence was 0.0020573968 .
 The Hessian was positive definite.
 Model rank = 35 / 35

Basis dimension (k) checking results. Low p-value (k-index<1) may indicate that k is too low, especially if edf is close to k'.

	k'	edf	k-index	p-value
s(position_field_x_cm,position_field_y_cm)	29.00	3.33	1.05	0.69

The QQ-plot indicates some departure from the normality assumption, as does the histogram of the residuals.

```
gam.check(gam.height_2022_07_05.num.log)
```



Method: GCV Optimizer: magic
Smoothing parameter selection converged after 7 iterations.
The RMS GCV score gradient at convergence was 1.3987332e-07 .
The Hessian was positive definite.
Model rank = 35 / 35

Basis dimension (k) checking results. Low p-value (k-index<1) may indicate that k is too low, especially if edf is close to k'.

	k'	edf	k-index	p-value
s(position_field_x_cm,position_field_y_cm)	29.00	3.64	1.06	0.54

The second model does not provide evidence of improvement over the previous model; therefore, we will retain the simpler model.

6.3.2 Smooth terms

We now want to check whether it is best to introduce the smooth terms interacting with each others (i.e. in a bivariate way, as above), in an additive way or in a linear manner.

We will now verify this assumption by starting with their additive introduction.

We need to adjust the number of basis in the `s()` function (the function estimating the smooth term) because otherwise the model cannot be fitted; this is done by modifying `k`, which sets the upper limit on the degrees of freedom associated with the `s()` smooth.

```

options(tinytex.verbose = TRUE)
k.x <- d.maize %>%
  pull(position_field_x_cm) %>%
  n_distinct()
k.y <- d.maize %>%
  pull(position_field_y_cm) %>%
  n_distinct()
gam.height_2022_07_05.num.add <- gam(height_2022_07_05.num ~
  s(position_field_x_cm,
    k = k.x - 1) +
  s(position_field_y_cm, k = k.y) +
  soil.fac +
  depth +
  seed.weight,
  data = d.maize)
summary(gam.height_2022_07_05.num.add)

```

Family: gaussian

Link function: identity

Formula:

```

height_2022_07_05.num ~ s(position_field_x_cm, k = k.x - 1) +
  s(position_field_y_cm, k = k.y) + soil.fac + depth + seed.weight

```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	191.4351675	38.8658065	4.9255	9.252e-06 ***
soil.facComposana	64.2187517	31.1835054	2.0594	0.04458 *
soil.facherbs	32.9455040	22.0411965	1.4947	0.14115
soil.facmixture	32.1973918	24.7165024	1.3027	0.19854
depth	0.0084514	3.6579231	0.0023	0.99817
seed.weight	-0.0699735	0.9779378	-0.0716	0.94324

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:

	edf	Ref.df	F	p-value
s(position_field_x_cm)	1.0000	1.000	0.9965	0.3229
s(position_field_y_cm)	1.9804	2.532	1.8673	0.1986

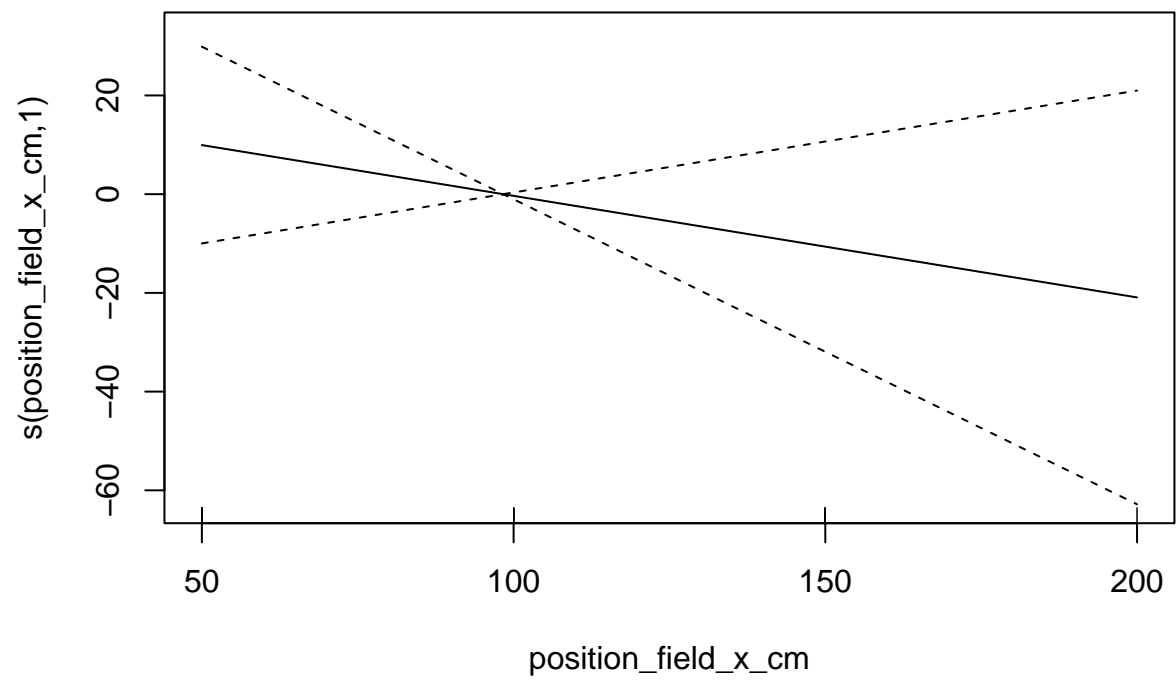
R-sq.(adj) = 0.0912 Deviance explained = 21.4%

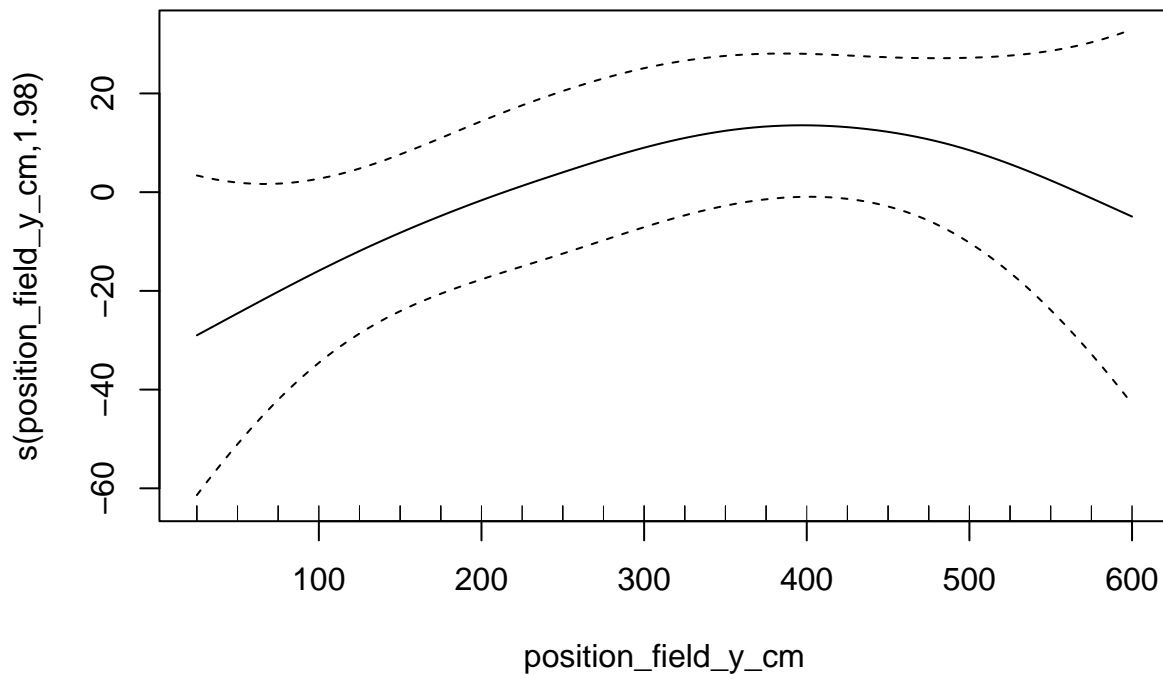
GCV = 2544.9 Scale est. = 2164 n = 60

```

plot(gam.height_2022_07_05.num.add)

```





We see that the variable *position_field_x_cm* takes the value 1.000 exactly, meaning that we can try to put this variable as linear in the model.

```
gam.height_2022_07_05.num.lin <- gam(height_2022_07_05.num ~
  position_field_x_cm +
  s(position_field_y_cm) +
  soil.fac +
  depth +
  seed.weight,
  data = d.maize)

##
summary(gam.height_2022_07_05.num.lin)
```

Family: gaussian

Link function: identity

Formula:

```
height_2022_07_05.num ~ position_field_x_cm + s(position_field_y_cm) +
  soil.fac + depth + seed.weight
```

Parametric coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	211.6950343	38.9200621	5.4392	1.526e-06 ***
position_field_x_cm	-0.2056887	0.2061204	-0.9979	0.3230
soil.facComposana	64.1898419	31.1880538	2.0582	0.0447 *
soil.facherbs	32.9033633	22.0479269	1.4924	0.1418

```

soil.facmixture      32.1752617  24.7175068  1.3017    0.1989
depth                0.0093235   3.6577769  0.0025    0.9980
seed.weight          -0.0703230   0.9779018 -0.0719    0.9430

```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Approximate significance of smooth terms:
```

```

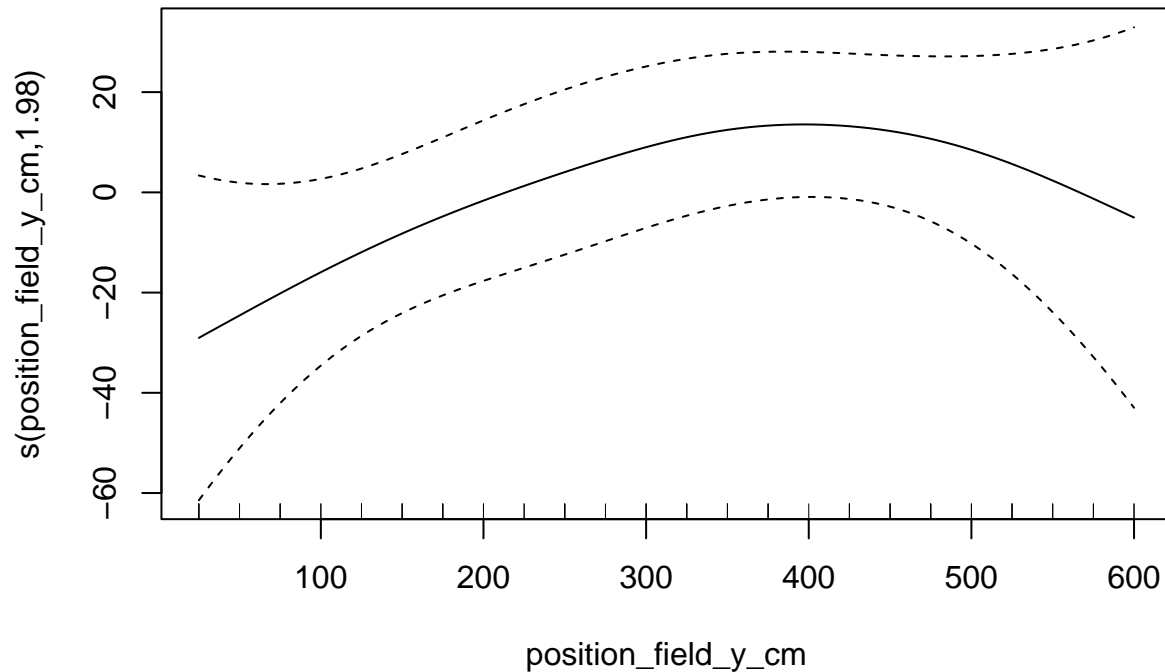
              edf Ref.df      F p-value
s(position_field_y_cm) 1.9785 2.5255 1.8754 0.1974

```

```
R-sq.(adj) = 0.0912   Deviance explained = 21.4%
```

```
GCV = 2544.7   Scale est. = 2163.9    n = 60
```

```
plot(gam.height_2022_07_05.num.lin)
```



We compare the models using the AIC and the BIC criteria.

We use this method instead of comparing the models using the `anova()` method because the models are not nested. Indeed, when fitting a GAM with smooth terms, if we modify these smooth terms, the smooths change, and as a result, the models are not nested.

```

cbind(AIC(gam.height_2022_07_05.num,
        gam.height_2022_07_05.num.add,
        gam.height_2022_07_05.num.lin),
      ##
      BIC(gam.height_2022_07_05.num,
          gam.height_2022_07_05.num.add,

```

```
gam.height_2022_07_05.num.lin))
```

	df	AIC	df	BIC
gam.height_2022_07_05.num	10.3341613	641.90103	10.3341613	663.54433
gam.height_2022_07_05.num.add	9.9804008	641.28752	9.9804008	662.18991
gam.height_2022_07_05.num.lin	9.9784850	641.28335	9.9784850	662.18174

Both criteria agree that the best model is the simpler one, where the variable *position_field_x_cm* is introduced in the model as a linear term.

In addition, we check whether the variable *soil.fac* has an influence on the response variable as a whole. Indeed, the above summary only shows the relative influence of each level compared to the reference level of *soil.fac*.

To achieve this result, we fit another model without this variable, and then we compare the two models using the `anova()` function.

We use the `anova()` function instead of the `drop1()` function, because the latter is not implemented for generalised additive models.

```
gam.height_2022_07_05.num.lin_reduced <- gam(height_2022_07_05.num ~
  position_field_x_cm +
  s(position_field_y_cm) +
  depth +
  seed.weight,
  data = d.maize)

##
anova(gam.height_2022_07_05.num.lin,
  gam.height_2022_07_05.num.lin_reduced, test = "Chisq")
```

Analysis of Deviance Table

Model 1: height_2022_07_05.num ~ position_field_x_cm + s(position_field_y_cm) +
soil.fac + depth + seed.weight

Model 2: height_2022_07_05.num ~ position_field_x_cm + s(position_field_y_cm) +
depth + seed.weight

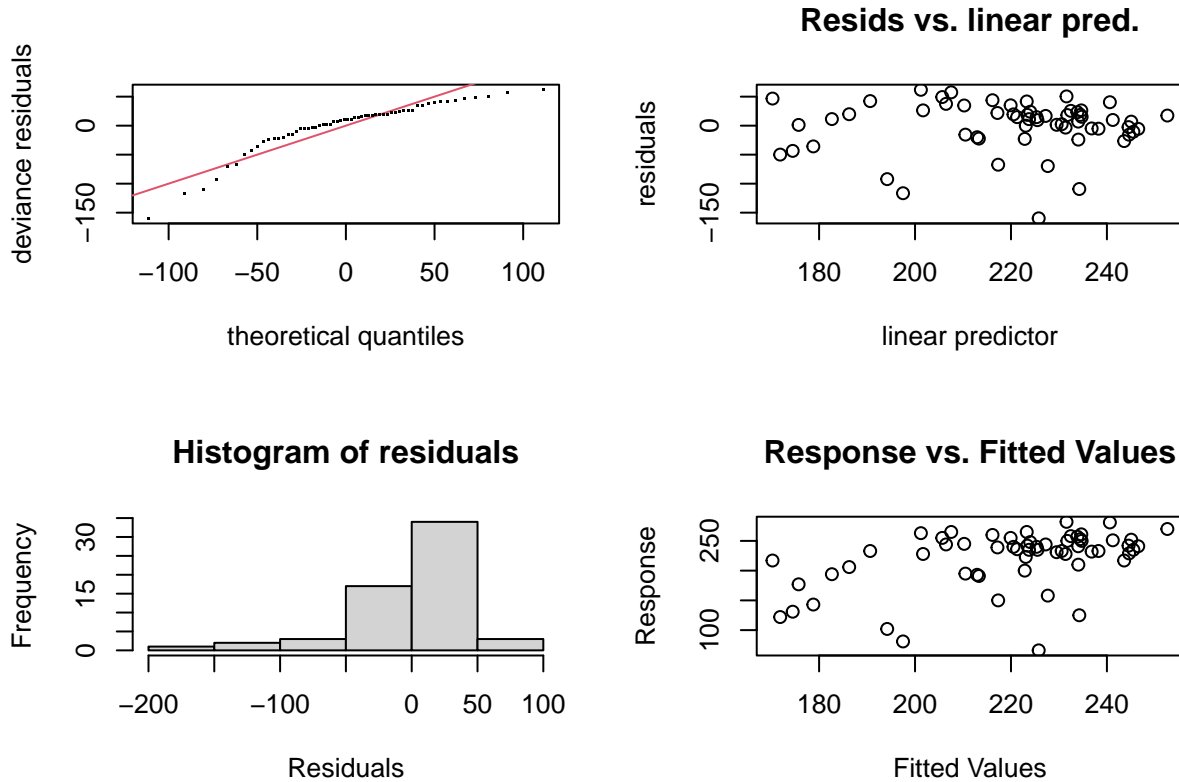
	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	50.4745	110405			
2	53.2103	117209	-2.73583	-6804.76	0.32567

The test shows that we have no evidence that including the variable *soil.fac* improves the model.

However, since the variable is in the design, we do not drop it from the model.

6.4 Model checking

```
gam.check(gam.height_2022_07_05.num.lin)
```



Method: GCV Optimizer: magic
Smoothing parameter selection converged after 5 iterations.
The RMS GCV score gradient at convergence was 0.0040526287 .
The Hessian was positive definite.
Model rank = 16 / 16

Basis dimension (k) checking results. Low p-value (k-index<1) may indicate that k is too low, especially if edf is close to k'.

	k'	edf	k-index	p-value
s(position_field_y_cm)	9.00	1.98	1.12	0.77

7 Methods description

To understand which factors influence the plant height, a generalised additive model was employed.

We used an additive approach because the explanatory variables *position_field_x_cm* and *position_field_y_cm* do not follow any clear distribution, thus they are introduced in the model as smooth terms.

The statistical analysis was performed using the R programming language, specifically version 4.3.1 (see citation below). The generalised additive model was fitted with the `gam()` function in the `{mgcv}` add-on package (see citation below).

Citations

```
citation()
```

To cite R in publications use:

```
R Core Team (2023). _R: A Language and Environment for Statistical
Computing_. R Foundation for Statistical Computing, Vienna, Austria.
<https://www.R-project.org/>.
```

A BibTeX entry for LaTeX users is

```
@Manual{,
  title = {R: A Language and Environment for Statistical Computing},
  author = {{R Core Team}},
  organization = {R Foundation for Statistical Computing},
  address = {Vienna, Austria},
  year = {2023},
  url = {https://www.R-project.org/},
}
```

We have invested a lot of time and effort in creating R, please cite it when using it for data analysis. See also 'citation("pkgname")' for citing R packages.

```
citation("mgcv")
```

2011 for generalized additive model method; 2016 for beyond exponential family; 2004 for strictly additive GCV based model method and basics of gamm; 2017 for overview; 2003 for thin plate regression splines.

Wood, S.N. (2011) Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society (B)* 73(1):3-36

Wood S.N., N. Pya and B. Saefken (2016) Smoothing parameter and model selection for general smooth models (with discussion). *Journal of the American Statistical Association* 111:1548-1575.

Wood, S.N. (2004) Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association*. 99:673-686.

Wood, S.N. (2017) *Generalized Additive Models: An Introduction with R* (2nd edition). Chapman and Hall/CRC.

Wood, S.N. (2003) Thin-plate regression splines. *Journal of the Royal Statistical Society (B)* 65(1):95-114.

To see these entries in BibTeX format, use 'print(<citation>, bibtex=TRUE)', 'toBibtex(.)', or set 'options(citation.bibtex.max=999)'.

8 Session information

```
sessionInfo()
```

```
R version 4.3.1 (2023-06-16)
```

```
Platform: aarch64-apple-darwin20 (64-bit)
```

```
Running under: macOS Sonoma 14.0
```

```
Matrix products: default
```

```
BLAS: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRblas.0.dylib
```

```
LAPACK: /Library/Frameworks/R.framework/Versions/4.3-arm64/Resources/lib/libRlapack.dylib; LAPACK vers
```

```
locale:
```

```
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
```

```
time zone: Europe/Zurich
```

```
tzcode source: internal
```

```
attached base packages:
```

```
[1] stats      graphics  grDevices  utils      datasets  methods    base
```

```
other attached packages:
```

```
[1] mgcv_1.8-42      nlme_3.1-162      multcomp_1.4-25    TH.data_1.1-2
```

```
[5] MASS_7.3-60      survival_3.5-5     mvtnorm_1.2-3       tibble_3.2.1
```

```
[9] ggplot2_3.4.4     kableExtra_1.3.4  dplyr_1.1.3         knitr_1.44
```

```
loaded via a namespace (and not attached):
```

```
[1] Matrix_1.6-1.1    gtable_0.3.4      compiler_4.3.1     webshot_0.5.5
```

```
[5] tidyselect_1.2.0  xml2_1.3.5         stringr_1.5.0      splines_4.3.1
```

```
[9] systemfonts_1.0.5 scales_1.2.1       yaml_2.3.7         fastmap_1.1.1
```

```
[13] lattice_0.21-8    R6_2.5.1           generics_0.1.3     munsell_0.5.0
```

```
[17] svglite_2.1.2     pillar_1.9.0       rlang_1.1.1        utf8_1.2.4
```

```
[21] stringi_1.7.12    xfun_0.40          viridisLite_0.4.2  cli_3.6.1
```

```
[25] withr_2.5.1       magrittr_2.0.3     digest_0.6.33      rvest_1.0.3
```

```
[29] grid_4.3.1        rstudioapi_0.15.0  sandwich_3.0-2     lifecycle_1.0.3
```

```
[33] vctrs_0.6.4       evaluate_0.22      glue_1.6.2         codetools_0.2-19
```

```
[37] zoo_1.8-12        fansi_1.0.5         colorspace_2.1-0    rmarkdown_2.25
```

```
[41] httr_1.4.7        tools_4.3.1        pkgconfig_2.0.3     htmltools_0.5.6.1
```