Generalized Linear Mixed Model with Spatial Covariates

by Alex Zolot (Zolotovitski)

StatVis Consulting

alex@zolot.us alexzol@microsoft.com

Introduction

- The task:
- Two Traits of subjects (plants) depends on
  1) Type (variable `Entry_Name`) and
  2) Location in 2D Fields (Field, Row, Column).
- Dependence of Type – fixed effect, on Location – random effect.
- All locations are different, but similarity decrease with distance.

Parts of Solution:

- Descriptive statistics and visualization.
- Data preparation.
- Building the model.
- Validation.
- Programming.
- Automation, GUI
- Optimization of experimental design

Building the Model.
Type – Location Decomposition

- If the attribute value collected on an experimental unit (cell) is represented by the term Y, then the attribute can be generally modeled as follows:

$$Y = T + L + Err\ .$$

- In general liner model (GLM) Y is linked to original variable Trait (Trait1 or Trait2) by linking function g() :

$$Y = g(Trait) \qquad (1)$$
$$Y = T + L + Err \qquad (2)$$

Box-Cox optimization

We looked for g() in form of Box-Cox transformation that maximize average by Entry_Name p-value of test Shapiro for normality.

The result of this procedure

Fun:	I	log(x)	x^1/3	sqrt(x)	x^2
Shapiro p.value:	0.37635	**0.52564**	0.49668	0.47207	0.17314

For simplicity we use $\lambda = 0$ corresponding to variable `Y=log(Trait)` that has almost highest normality, but easier for understanding.

- Tests for homoschedastisity also confirmed advantage of logarithmic linking function in glm.
- So in our program we use log linking Y = log(Trait) with following variables names:

Tra = Trait1 or Trait2 (3)
LTra = Y = log(Trait)

- with type – location decomposition

Y = Y_ty + Y_loc + res (4)
Tra = Tra_ty * Tra_loc + noise

- where

Tra_ty = exp(Y_ty) and Tra_loc = exp(Y_ loc)

- In our case type "ty" is related to variable Entry_Name and location "loc" to tuple (Testing_Site, Field, Row, Column) .

Iteration of Type – Location decomposition.

To get decomposition (2), we use the following iterative procedure:

$$Y = Y(\text{type}, \text{loc}) = Y0 = \log(\text{Trait})$$

Do until convergence: Y_old = Y

T(type) = mean(Y | Type = type) , where Type = EntryName

$$L0 = Y - T(\text{type})$$

For each TSF, using `krige.cv` package gstat :

```
L(loc) = cv.Predict (Krig(L0 ~ Row + Column, loc, θ))

Y_new= Y0 - L(loc)

Y = (1 - λ ) * Y_old + λ  * Y_new
```

Loop until `||Y_new - Y_old|| < ε`

```
T(type) = mean( Y | Type = type)
```

where θ is the set of parameters of kriging that we have to optimize, and λ is parameter of acceleration.

- We control SSE (sum of squares of residuals) and after it differences becomes smaller than tolerance or after fixed number of "burn out" cycles we get mean and standard deviation of Y_loc and Y_ty:

$$Y\_loc.m = mean(Y\_loc \mid burnOut < iter \leq maxiter)$$

$$Y\_loc.sd = sd(Y\_loc \mid burnOut < iter \leq maxiter)$$

- Residuals depend on Row, Column after excluding Type and Test_Site components:

```
 library(nlme)
 fm1 <- lme(LTra ~ Entry_Name, sds, random = ~ 0 |   Entry_Name)
  #effect of  Testing_Site  =======
  sds$resid1= fm1$resid[,1]     # now means by Entry_Name are excluded
  fm2 <- lme(resid1 ~ TSF, sds, random = ~ 0 | TSF)    # not
necessary, just to exclude mean by TSF.
  sds$resid2= fm2$resid[,1]     # now means by TSF are excluded
```

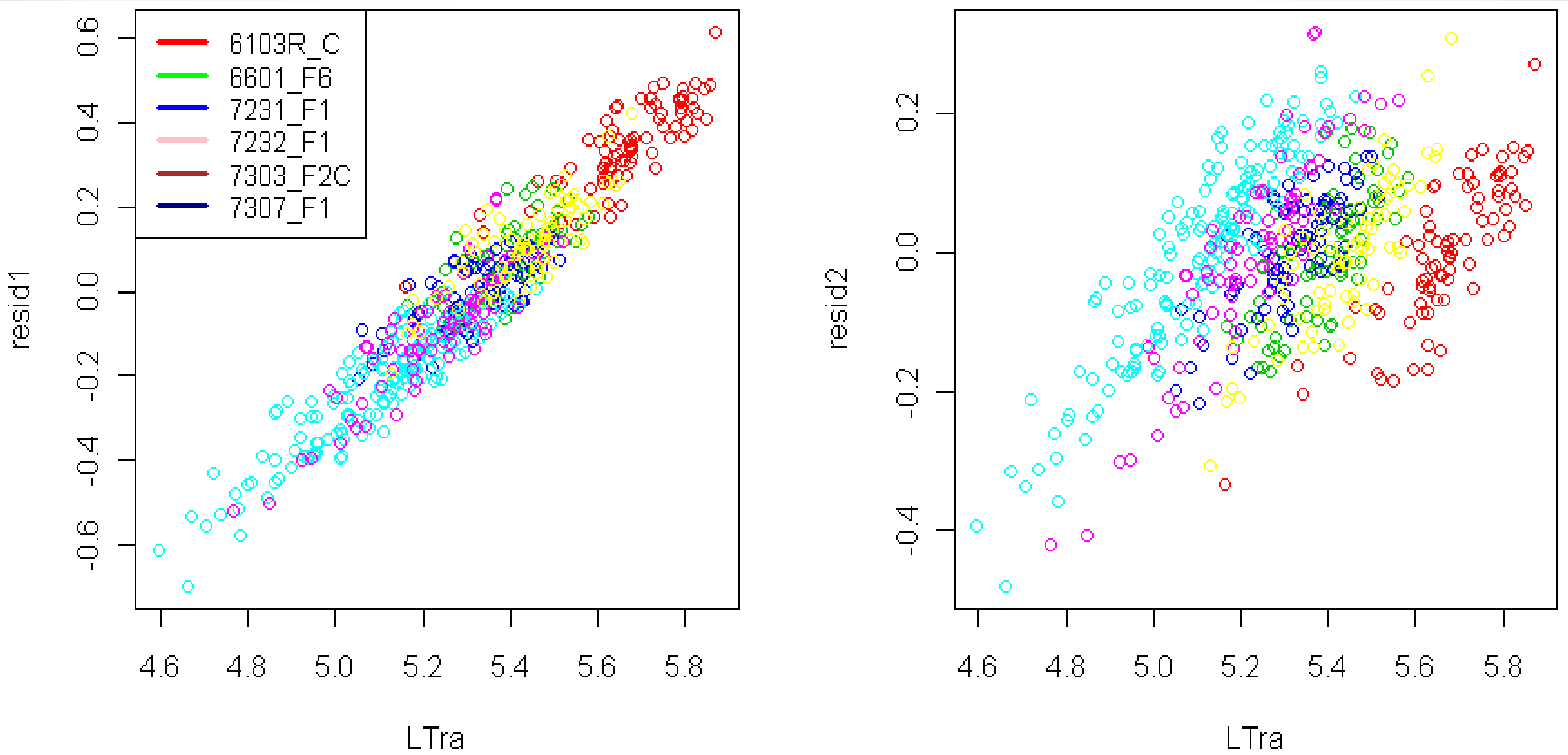

Fig.2. Excluding Type-dependence in 0- approximation.

Kriging cross-validation and optimization.

- Two kriging parameters – range and nugget
- Methods of Nelder and Mead (1965)
- Optimization of kriging parameters is very important and time-consuming procedure, so our results must be considered as preliminary.
- Linear regression on residuals with predictors Row and Column, that we considered as numerical variables – so all our prediction on this stage used only 4 kriging adjustment parameters – sill, range, nugget, and anisotropy.

- We also tried to use regression with Row and Column as random effects, but found that additional degrees of freedom increase AIC:

```
ds$cRow=paste('r',ds$Row, sep='')
  ds$cCol=paste('c',ds$Column, sep='')

  lm00= glm( resid2 ~ var1.pred,                                    data = ds)
  lm0=  glm( resid2 ~ var1.pred + Column + Row  ,                   data = ds)
  lmR=  glm( resid2 ~ var1.pred + Column + Row + cRow ,             data = ds)
  lmC=  glm( resid2 ~ var1.pred + Column + Row + cCol ,             data = ds)
  lmRC= glm( resid2 ~ var1.pred + Column + Row + cCol+ cRow ,       data = ds)

   c(AIC(lm00), AIC(lm0), AIC(lmC), AIC(lmR), AIC(lmRC))
#  -3615.188   -3611.492 -3584.912 -3584.497 -3568.149
```

- Kriging on residuals after excluding Type effect in 0-approximation:

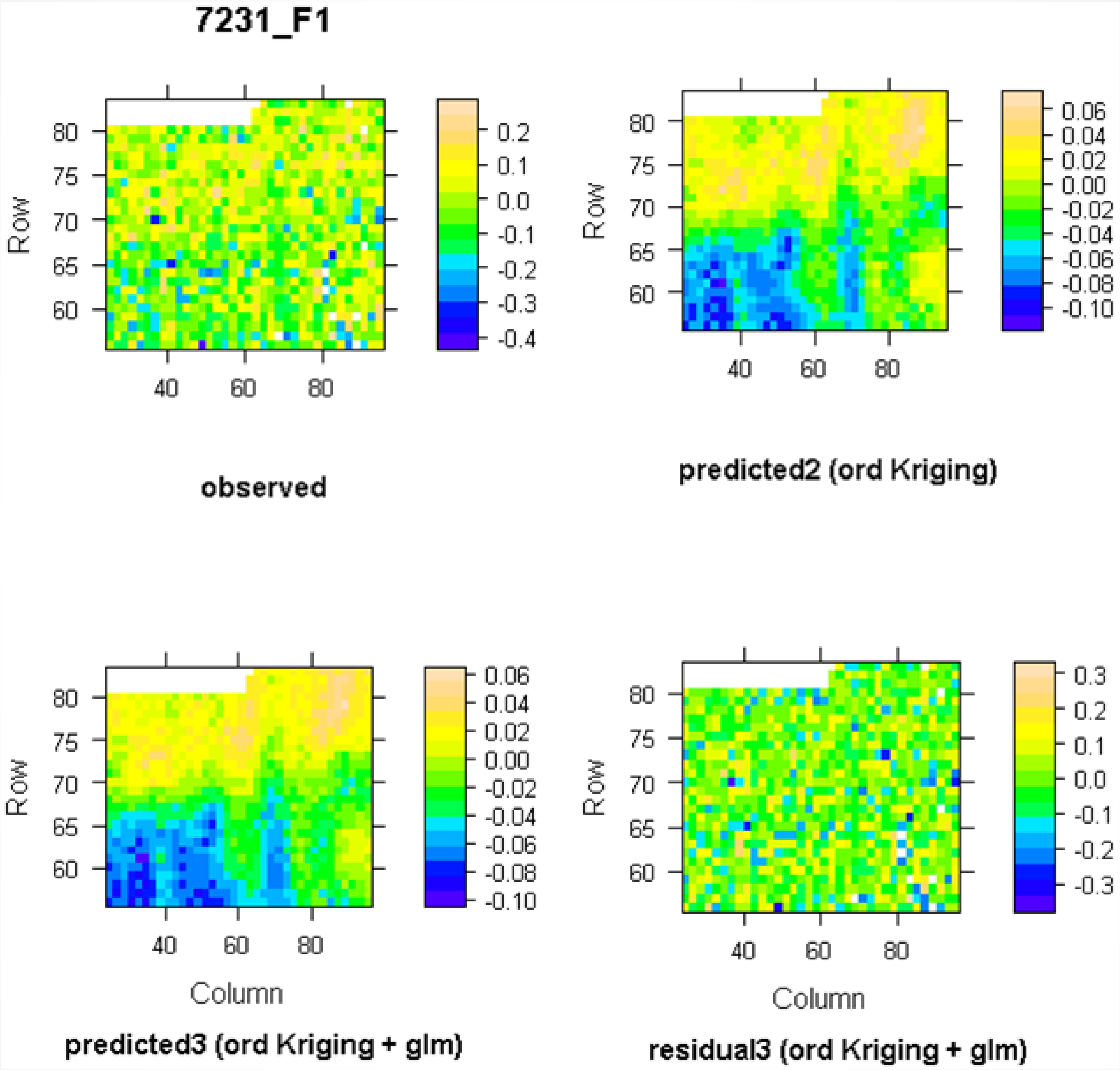

Fig.4. Result of kriging + glm for TSF = 7231_F on Loc – dependant part of data

Variograms and anisotropy

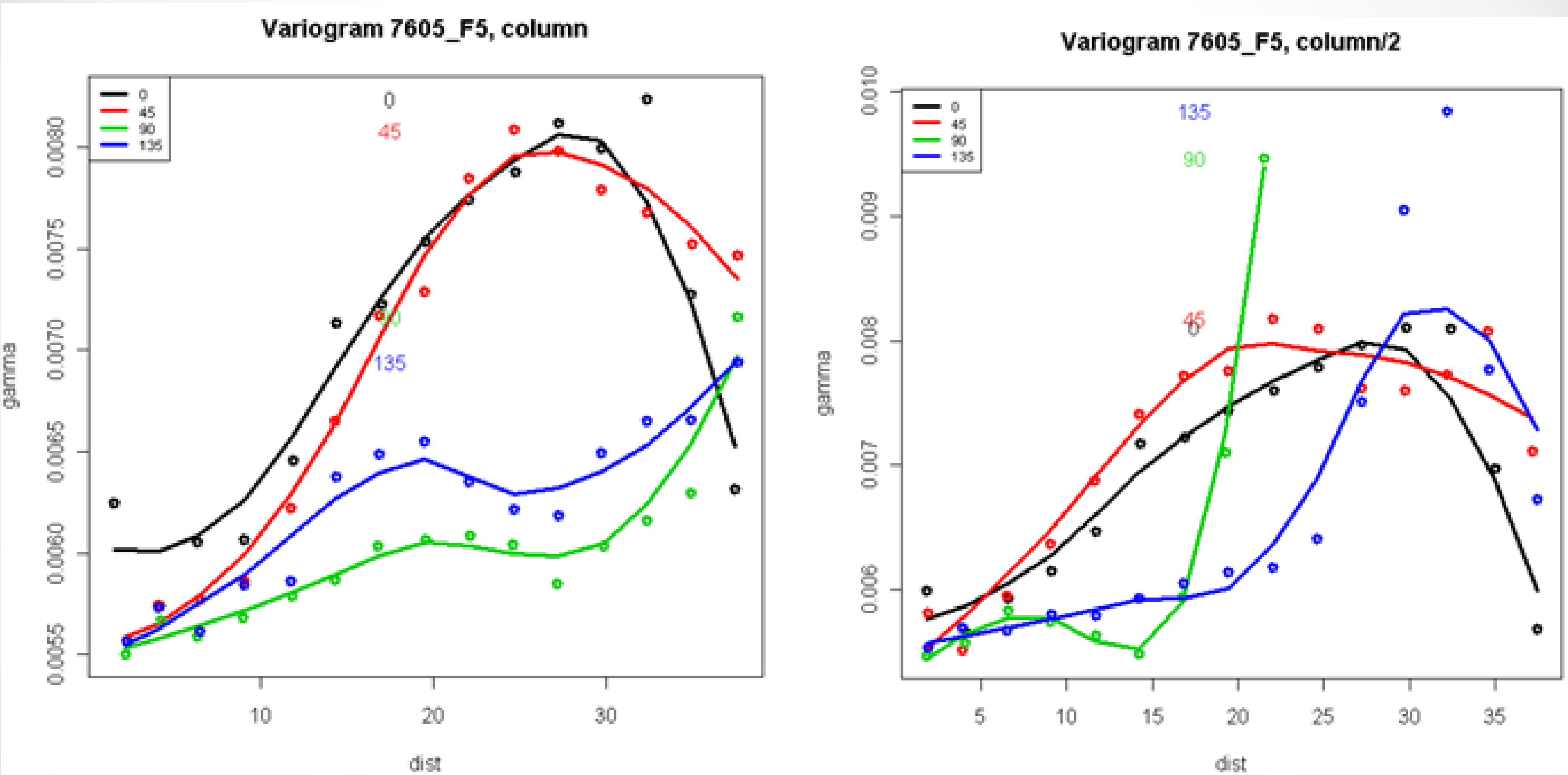

Fig 6. Variograms for different angles for TSF = 7605_F5.

- From Fig.7 we see that elliptical model

variogram (diffRow, diffColumn) = f ((diffRow /a)^2 + (diffColumn /b)^2)

with one parameter of anisotropy

anis = b / a

is not very good fitting for anisotropy but in standard kriging procedures only this model of anisotropy is implemented. To improve accuracy of our model in future we could use a multistep approach to overcome this inaccuracy of elliptical model.

Choosing number of iterations.

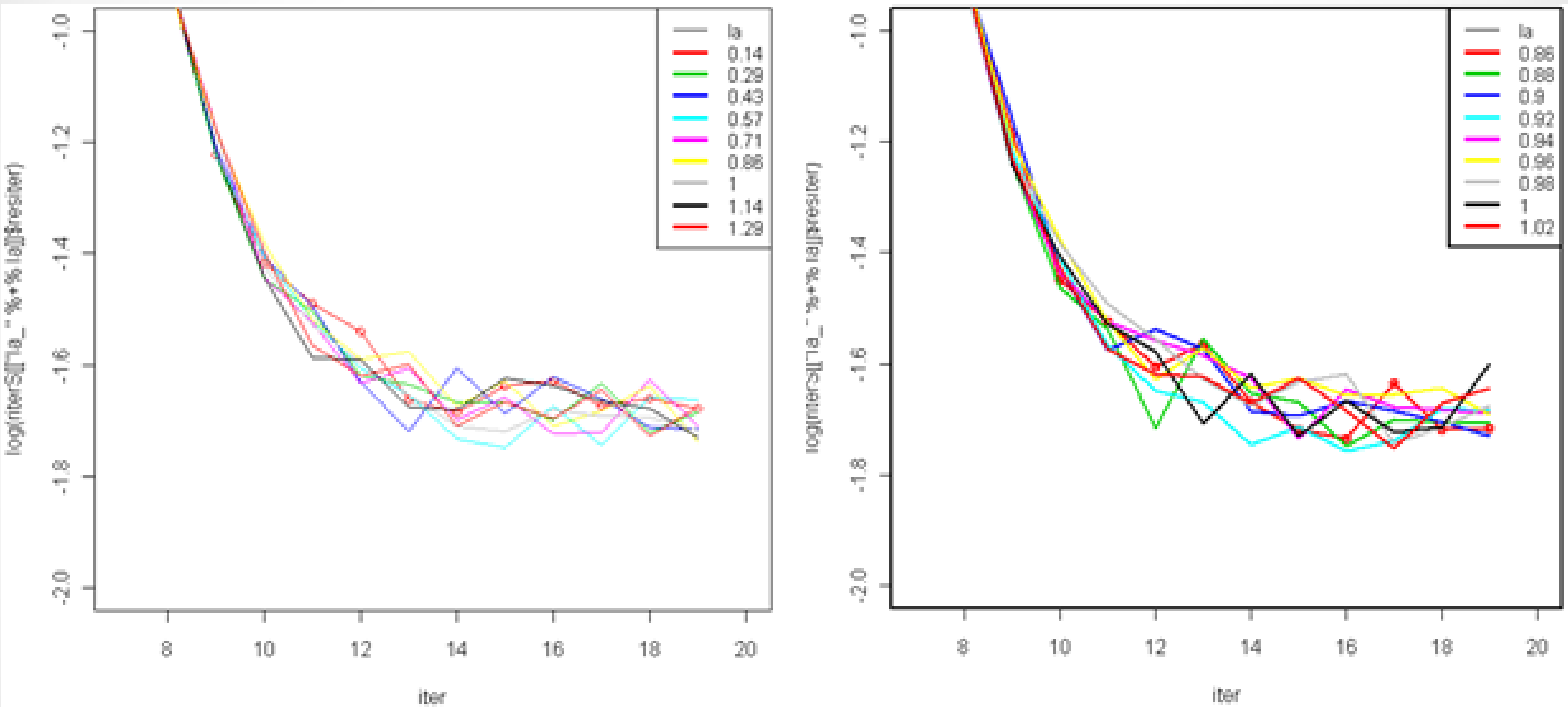

Fig. 10. ln(SSE) vs iteration.for different acceleration parameter la = λ.

- As a result sharpness of signal increased essentially:

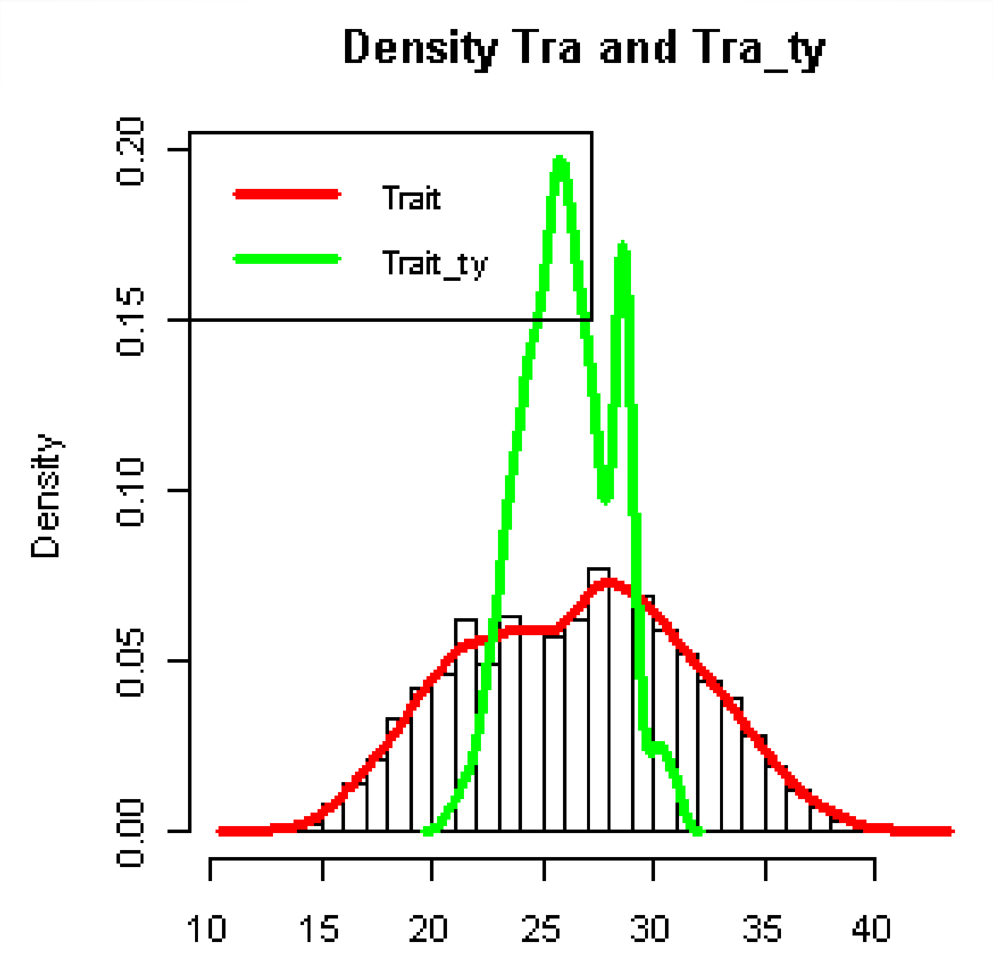

Fig.9. Density for distribution Tra and Tra_Ty for Trait=1, Treatment =2.

Programming. Automation, GUI

- R for analyzing and modeling
- packages 'stats', 'sqldf' , 'spatstat' , 'gstat' , 'sp', 'lattice', 'tcltk', 'tkrplot', 'graphics'

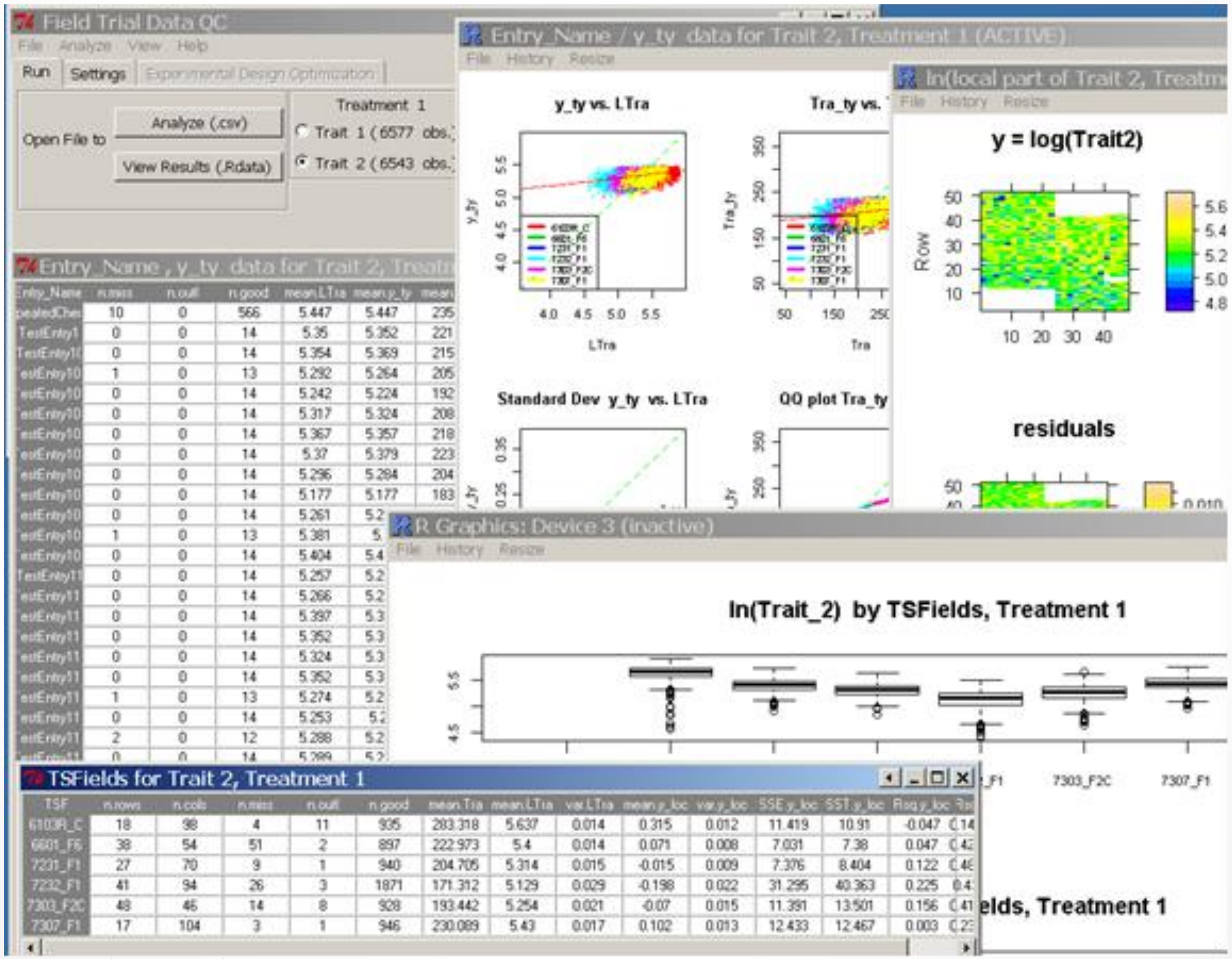

Fig.15.Screenshot of GUI

Conclusion

- Noise: The sum of the squared residuals of the model should be minimized.

  Resulting SSE:

Dataset 1:

Treatment	Trait	SSE	SST	Rsq
1	1	14.308	146.087	0.902
1	2	48.392	191.456	0.747

Dataset 2:

Treatment	Trait	SSE	SST	Rsq
1	1	40.769	286.499	0.858
1	2	80.945	317.27	0.745
2	1	35.998	150.875	0.761
2	2	58.341	175.262	0.667

- Parsimony: Fitted parameters for location-based artifacts must comprise a relatively small portion of the total number of parameters.

    We used only 4 fitting parameters of kriging for each (Treatment, Trait, TSField)

- Signal: The remaining signal in the dataset should be maximized, as measured by a statistical test to differentiate the entries.

    Sharpness of signal increased essentially, as Fig.8-9 shows.

- Dropped values: The amount of dropped data values should be kept to a minimum.

    We dropped about 1% as outliers.

- Speed and ease of use: Some automation with an intuitive user-interactive interface.

    Our GUI has only 6 buttons in Tcl/Tk and only one button in RExcel.

Next Steps

- The performance could be improved essentially if we combine iterations with cross-validation. Results that we delivered were obtained with 20 fold cross-validation and 19 iterations, that means dataset was scanned 20 * 19 = 380 times and it took about 154 min. If we combine iterations with cross-validation, we estimate to reach the same accuracy in about 40 scans, that is 10 time faster, so it would take less than 1 min.

- We can also improve accuracy by using two stage kriging to extend managing of anisotropy in our variogram model from 1-parameter ellipse with main axis in column direction to at least 2-parameters of two ellipses in column and row direction or in arbitrary angle. We estimate possible accuracy improvement in about 25- 30% decrease of SSE.

Generalized Linear Mixed Model with Spatial Covariates

by Alex Zolot (Zolotovitski)

alex@zolot.us www.zolot.us