Visualization of Outlier Diagnostic Model in Data Space: Boundaries and Performance

Wenjing Wang[1], Dianne Cook[2], Earo Wang[2]
[1]Renmin University of China , [2]Monash University

0.1 Introduction

Explore how the algorithm works. Wickham, Cook, and Hofmann (2015) pointed out whenever we can gain insight into the process of model fitting, we should. Observing interations helps us understand how the algorithm works and real potential pitfalls. Developing suitable visualizations forces us to think deeply about exactly what the algorithm does and can suggest possible avenues for improvement.

It is difficult to understand statistical models in high-dimensional space. Visualizing the model in data space as a means to better understand of the model fit. When a linear regression model consist of a single continuous predictor and a single response, the fitted model is a simply visualized as a line in two dimensions. When a model involves two predictors it may be visualized as a surface in three dimensions.

Outlier diagnostic models are widely used for detecting influential points in regression models. We are interested in the following questions:

(1) Where is the boundaries of `normal` and `outlier` in data space produced by the diagnostic models?

(2) How to judge the performance of different diagnostic models?

(3) In non-linear regression case where the influential points are not determined `distance`, how can we locating them?

0.2 Fitting Quantile Regression: Algorithm and Data

The fitting method of quantile regression is different from ordinary regression model. We provide visualization method to understand the most widely used algorithms: simplex method and internal points methods.

Three formulations of quantile regression (QR) at level τ

$$\begin{aligned}
&\min \frac{1}{n}\sum_{i=1}^{n}\rho_\tau(y_i - x_i\beta) \\
&\min(\tau - \frac{1}{2})(\bar{y} - \bar{x}\beta) + \frac{1}{2n}\sum_{i=1} n|y_i - x_i\beta| \\
&\textstyle\sum_{i=1}^{n} y_i a_i \quad s.t. \quad X^T a = (1-\tau)X^T 1_n \quad and \quad a \in [0,1]^n
\end{aligned} \tag{1}$$

where $\rho_\tau(t) = \tau t^+ + (1-\tau)t^-$

Linear programming are expressed as:

$$\begin{aligned}
&\text{min} c^T z \\
&s.t. \quad Az = b \quad z \geq 0
\end{aligned} \tag{2}$$

We fitting quantile regression model by casting quantile regression as linear programming problem

$$\begin{aligned}
z &= (\beta^+ \quad \beta^- \quad \xi^+ \quad \xi^-)^T \\
c &= (0 \quad 0 \quad \tau/n \quad (1-\tau)/n)^T \\
A &= (X \quad -X \quad I \quad -I) \\
b &= Y
\end{aligned} \tag{3}$$

where c and z are m-vectors with $m = 2p + 2n$, $\boldsymbol{A}$ is a n-by-m matrix, and $\xi = Y - X\beta$ is the residual vector.

simplex theory

Let $B \equiv B_1, ..., B_n \subsetneqq 1, .., n$ denote an index set. $A_B \equiv [A_{B_1}, ...A_{B_n}]$ denote an invertible sub-matrix of A. $z^* \equiv [z_1^*, ..., z_m^*]$ is called a basic solution if z^* satisfies:

$$\begin{aligned}
z_B^* &= A_B^{-1} b \\
z_j^* &= 0 \quad for \quad j \in 1, ..., m
\end{aligned} \tag{4}$$

z^* is an optiomal solution if,

$$\begin{aligned}
z_B^* = A_B^{-1} b \geq 0 \\
c - A^T A_B^{-1} c_B \geq 0
\end{aligned} \tag{5}$$

Simplex tableau is:

$$\begin{pmatrix} -c_B^T z_B & c^T - c_B^T A_B^{-1} A \\ A_B^{-1} b & A_B^{-1} A \end{pmatrix}$$

As the algorithm show, we first find the set B which consist the points solved by linear system of equations.

```r
library(quokar)
library(quantreg)
```

```
## Loading required package: SparseM

##
## Attaching package: 'SparseM'

## The following object is masked from 'package:base':
##
##     backsolve
```

```
library(ggplot2)
library(gridExtra)
library(tidyr)
library(dplyr)
```

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

## The following object is masked from 'package:gridExtra':
##
##     combine

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

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

```
data(ais)
ais_female <- dplyr::filter(ais, Sex == 1)
```

```
tau <- c(0.1, 0.5, 0.9)
br <- rq(BMI ~ LBM, tau = tau, data = ais_female, method = "br")
coef <- br$coef
br_result <- frame_br(br, tau)
```

```
## [1] "Observations used in br method fitting"
##   tau_flag indice1 indice2
## 1  tau=0.1      84      86
## 2  tau=0.5      15      97
## 3  tau=0.9      71      29
```

```
origin_obs <- br_result$all_observation
use_obs <- br_result$fitting_point
```

In multi-variable case: we imply interactive visualization

```
## multi-variable case
br <- rq(BMI ~ LBM + Bfat , tau = tau, data = ais_female, method = 'br')
tau <- c(0.1, 0.5, 0.9)
br_result <- frame_br(br, tau)
```

```
## [1] "Observations used in br method fitting"
##   tau_flag indice1 indice2 indice3
## 1  tau=0.1      50       4      47
## 2  tau=0.5      46      98      89
## 3  tau=0.9      53      74      67
```

0.2.0.1 Better Method for Visualization

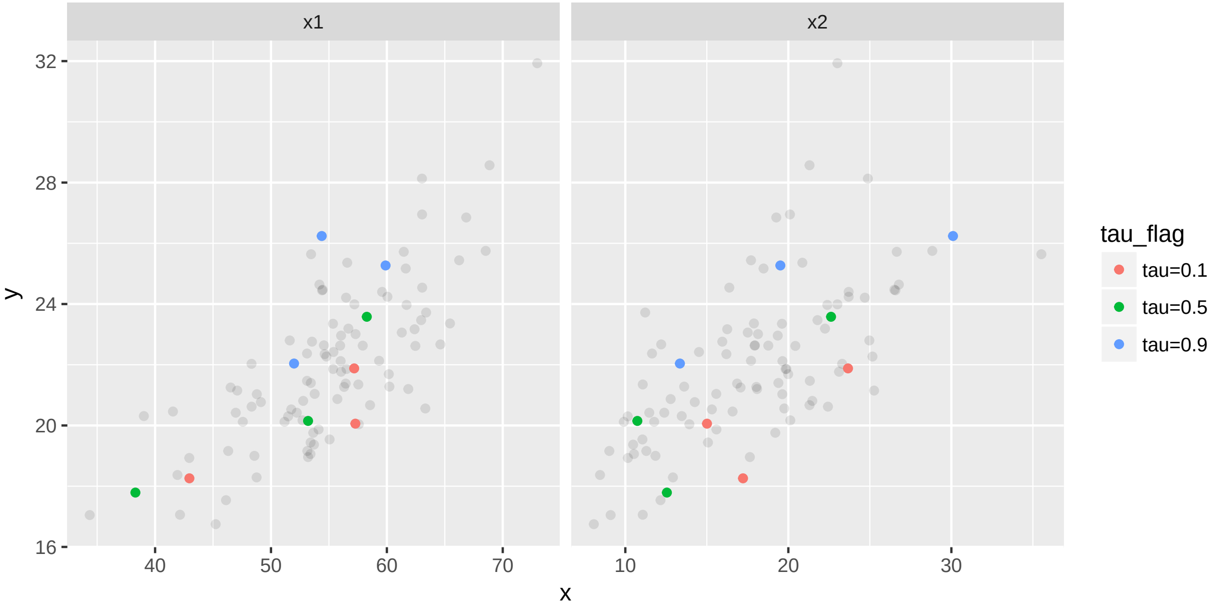

Figure 1: Simplex algorithhom in multi responses model.

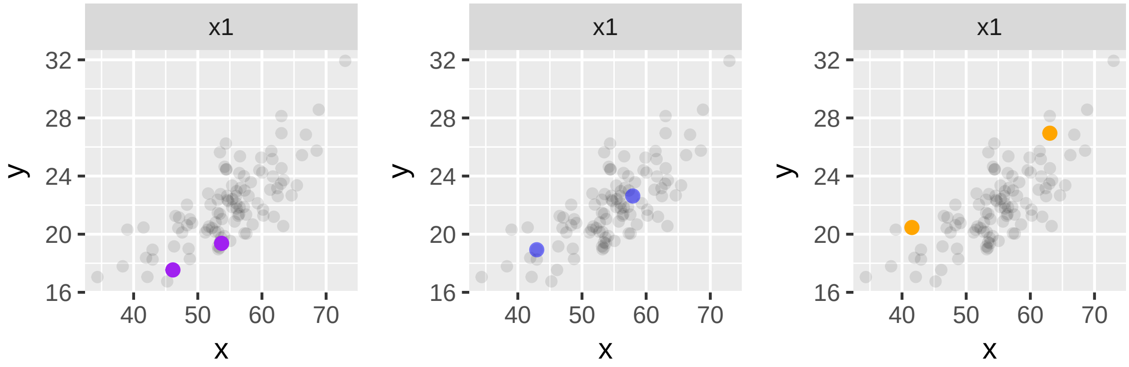

Figure 2: Simplex algorithom in multi responses model.

0.2.1 Interior Points Methods: Weighting

```
##              tau0.1       tau0.5       tau0.9
## [1,] 3.463266e-10 8.032508e-11 1.294102e-09
## [2,] 3.676714e-08 1.552504e-10 2.581219e-09
## [3,] 3.680850e-11 1.659972e-08 9.871077e-09
## [4,] 5.848082e-11 1.532711e-09 6.737070e-09
## [5,] 5.315053e-09 9.057593e-11 2.478669e-09
## [6,] 5.902536e-11 1.364510e-09 7.639299e-09
```

```
tau <- c(0.1, 0.5, 0.9)
fn <- rq(BMI ~ LBM + Bfat, data = ais_female, tau = tau, method = 'fn')
fn_obs <- frame_fn_obs(fn, tau)
head(fn_obs)
```

```
##              tau0.1       tau0.5       tau0.9
## [1,] 1.082709e-09 1.517760e-14 6.307261e-16
## [2,] 2.202128e-09 2.358017e-14 9.430150e-16
## [3,] 1.017783e-09 4.585456e-13 3.516256e-15
```

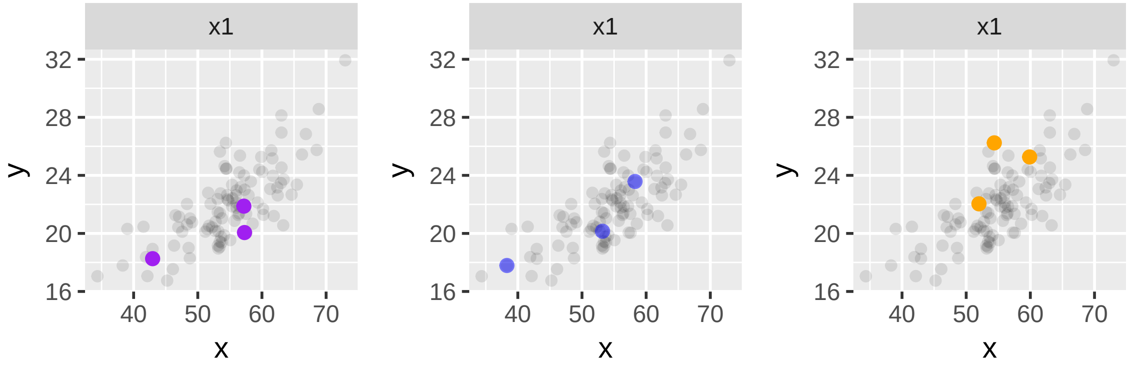

Figure 3: Simplex algorithom in multi responses model.

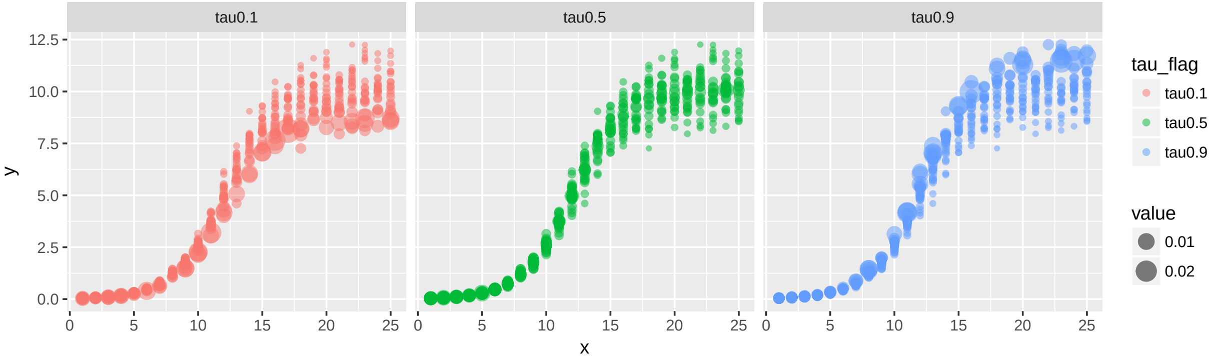

Figure 4: Simplex algorithom in multi responses model.

```
## [4,] 1.576856e-01 5.754519e-14 1.602429e-15
## [5,] 1.852327e-09 2.436167e-14 1.118007e-15
## [6,] 5.858695e-10 3.150193e-12 5.205425e-15
```

0.2.2 Non-linear case

0.3 General Framework of Visualizing Outlier Diagnostic Models

0.4 Linear Quantile Regression Outlier Diagnostic

We used ais data as example to do our anaylysis.

0.5 Non-linear Quantile Regression Outlier Diagnostic

Visualizing Outlier Diagnostic Models for Mix Level Model

Wickham, Hadley, Dianne Cook, and Heike Hofmann. 2015. “Visualizing Statistical Models: Removing the Blindfold.” *Statistical Analysis and Data Mining: The ASA Data Science Journal* 8 (4). Wiley Online Library: 203–25.

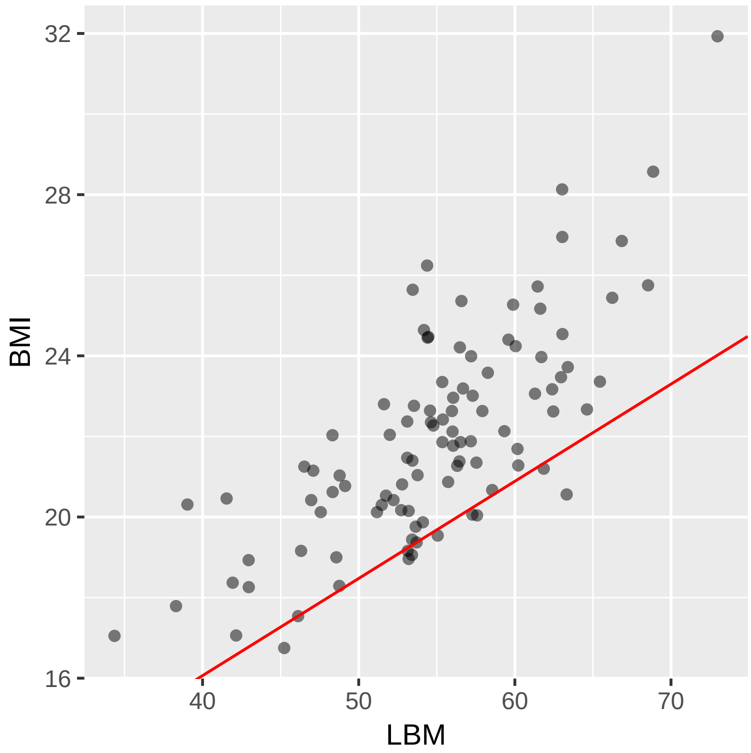

Figure 5: data and model.

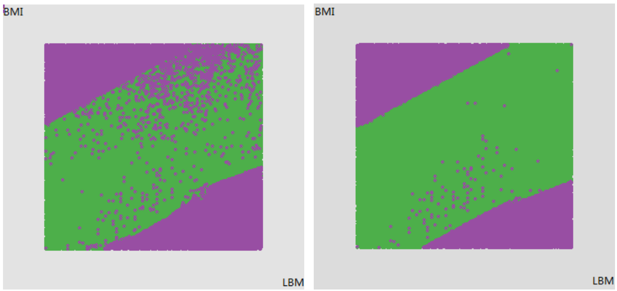

Figure 6: General cook distance and Q function outlier diagnostic for single variable quantile regression models. In single variable case, we get 2-dimension result with purple and green points representing outlier and normal.

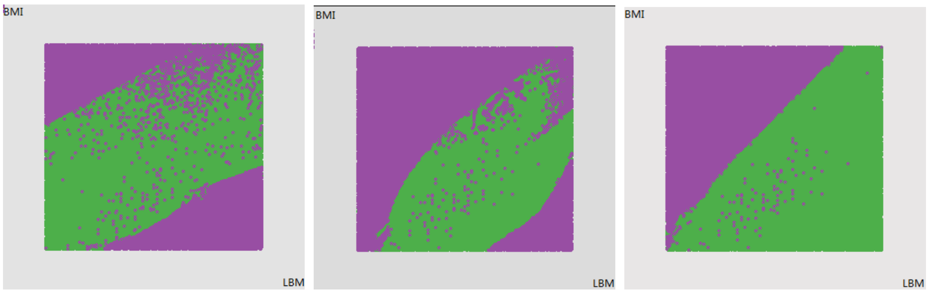

Figure 7: General cook distance outlier diagnostic for single variable quantile regression models. In single variable case, we get 2-dimension result with purple and green points representing outlier and normal.

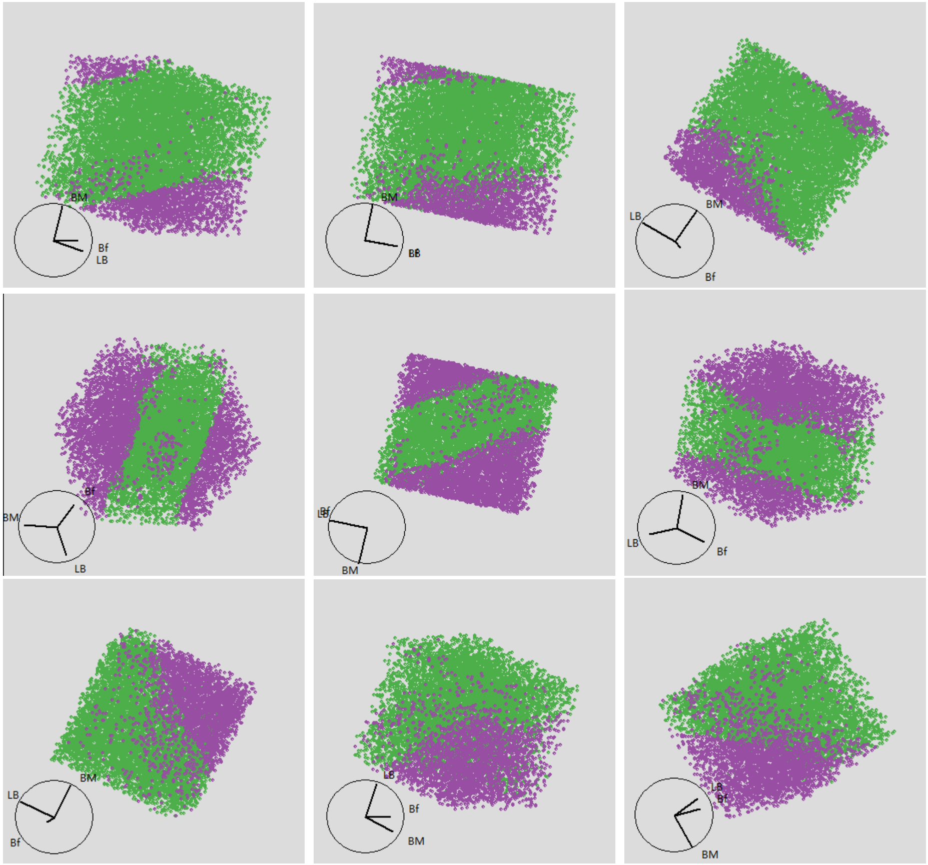

Figure 8: General cook distance outlier diagnostic for two response variables quantile regression models. In multi-variable case, we get 3-dimension result with purple and green points representing outlier and normal.

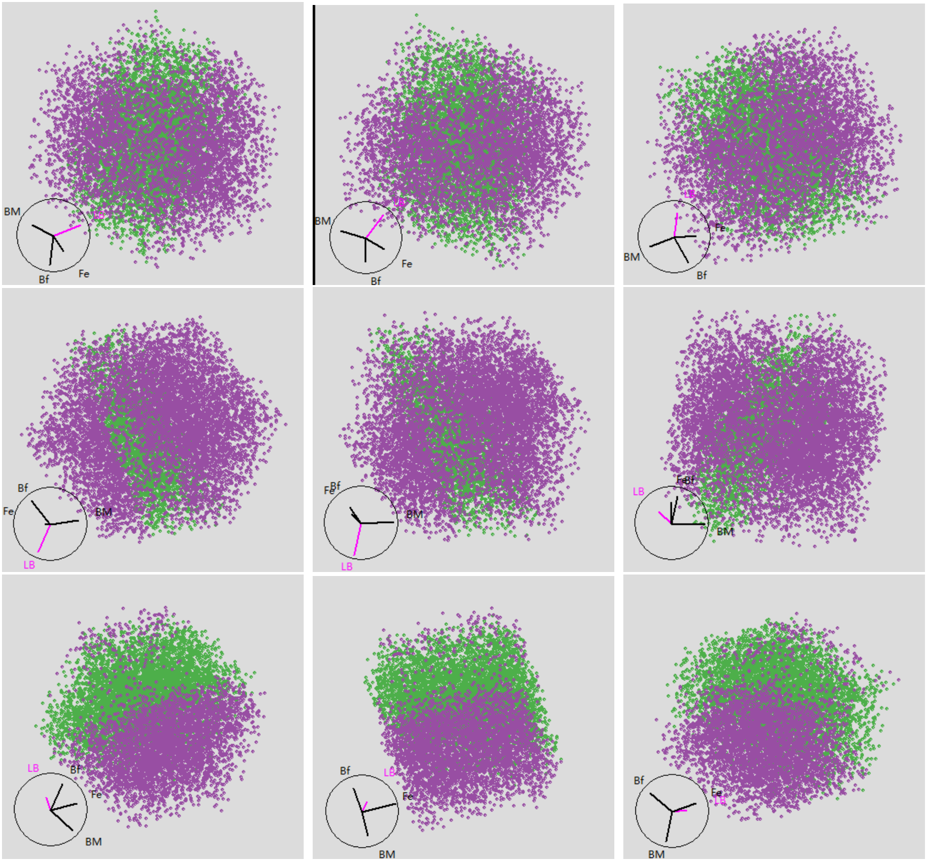

Figure 9: General cook distance outlier diagnostic for three response variables quantile regression models. In multi-variable case, we get 3-dimension result with purple and green points representing outlier and normal.

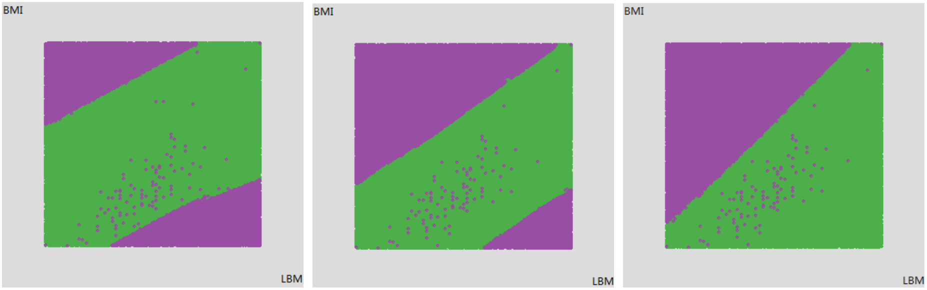

Figure 10: Q function outlier diagnostic for one response variables quantile regression models. In single-variable case, we get 2-dimension result with purple and green points representing outlier and normal.

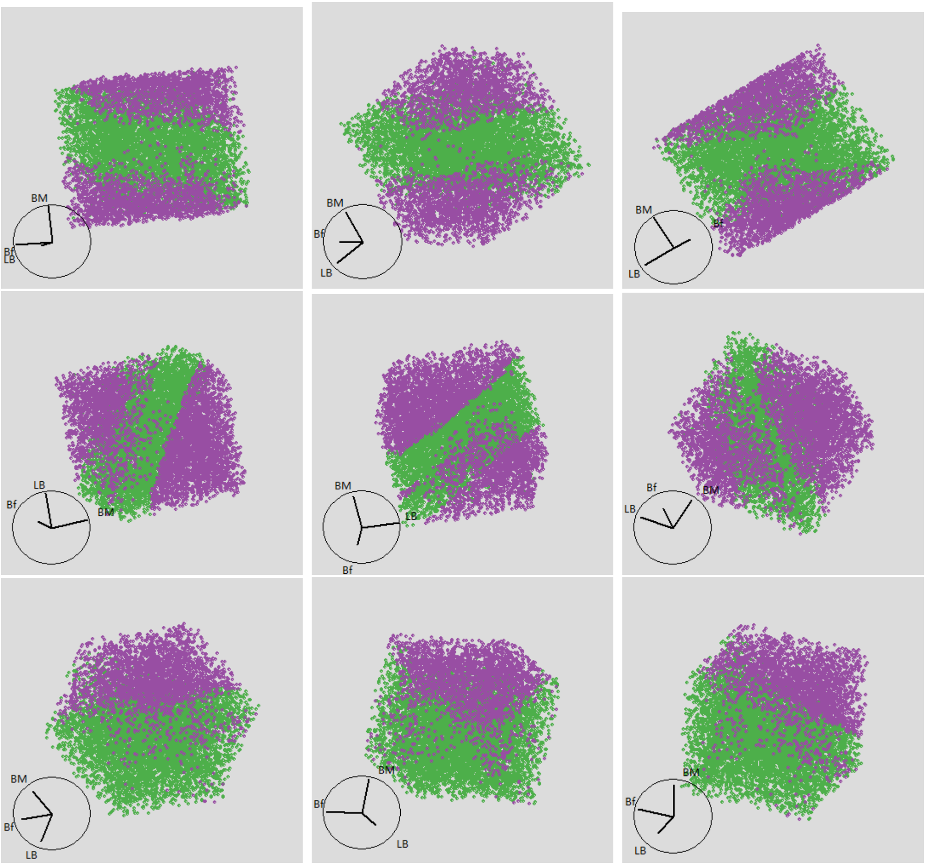

Figure 11: Q function outlier diagnostic for two response variables quantile regression models. In multi-variable case, we get 3-dimension result with purple and green points representing outlier and normal.

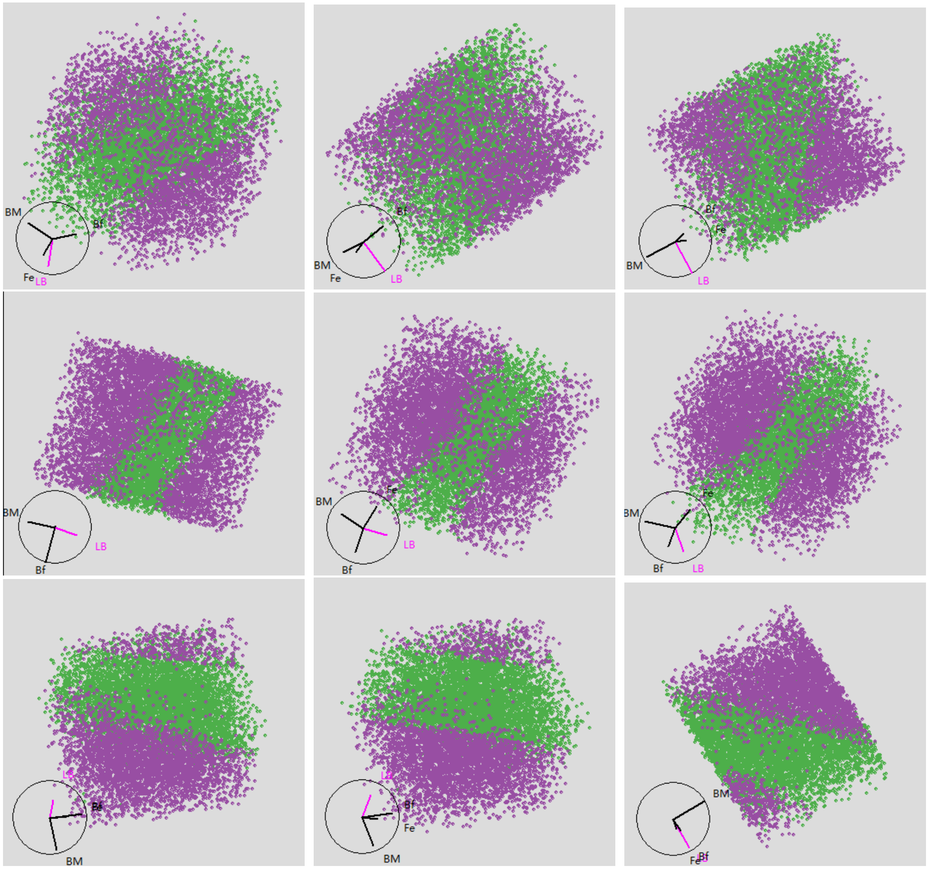

Figure 12: Q function outlier diagnostic for three response variables quantile regression models. In multi-variable case, we get 3-dimension result with purple and green points representing outlier and normal.