Food Demand Forecast

Business, Economic and Financial Data Project

Pierpaolo D'Odorico, Massimiliano Conte and Eddie Rossi

Modelling

Since we want to organize the goods for each specific center, we need to forecast the demand for each specific center.

We also need to stratify for each unique meal, since each of them requires a different set of raw materials. We propose a two-stage approach:

- First we account for the temporal relationship using the **linear model**, obtaining (hopefully) i.i.d. residuals.
- Then we model the **obtained residuals**, using some flexible method such as the **gradient boosting**.

Linear model

We want to fit a straight line, between demand and time, for each combination of center and meal.

This mean we should fit $N^o centers \cdot N^o meals = 77 \cdot 51 = 3927$ linear models. But if we carefully craft some indicator variables we can specify all the simple linear models in to one **single big linear model**.

Linear model

$$Y_{ij} = \beta_{0ij} + \beta_{1ij} week + \mathcal{E}_{ij}$$
$$\forall i = 1, ..., 77; j = 1, ..., 51$$

Is equivalent to:

$$Y = \beta_0 + \beta_1 week + X_{ind}\beta_{level} + X_{ind}\beta_{slope} \cdot week + \mathcal{E}$$

Linear model

X_{ind} is a vector with $51 \cdot 77 - 1 = 3926$ columns, and is obtained as the interaction between the **dummy expansion** of the categorical variables center_id and meal_id.

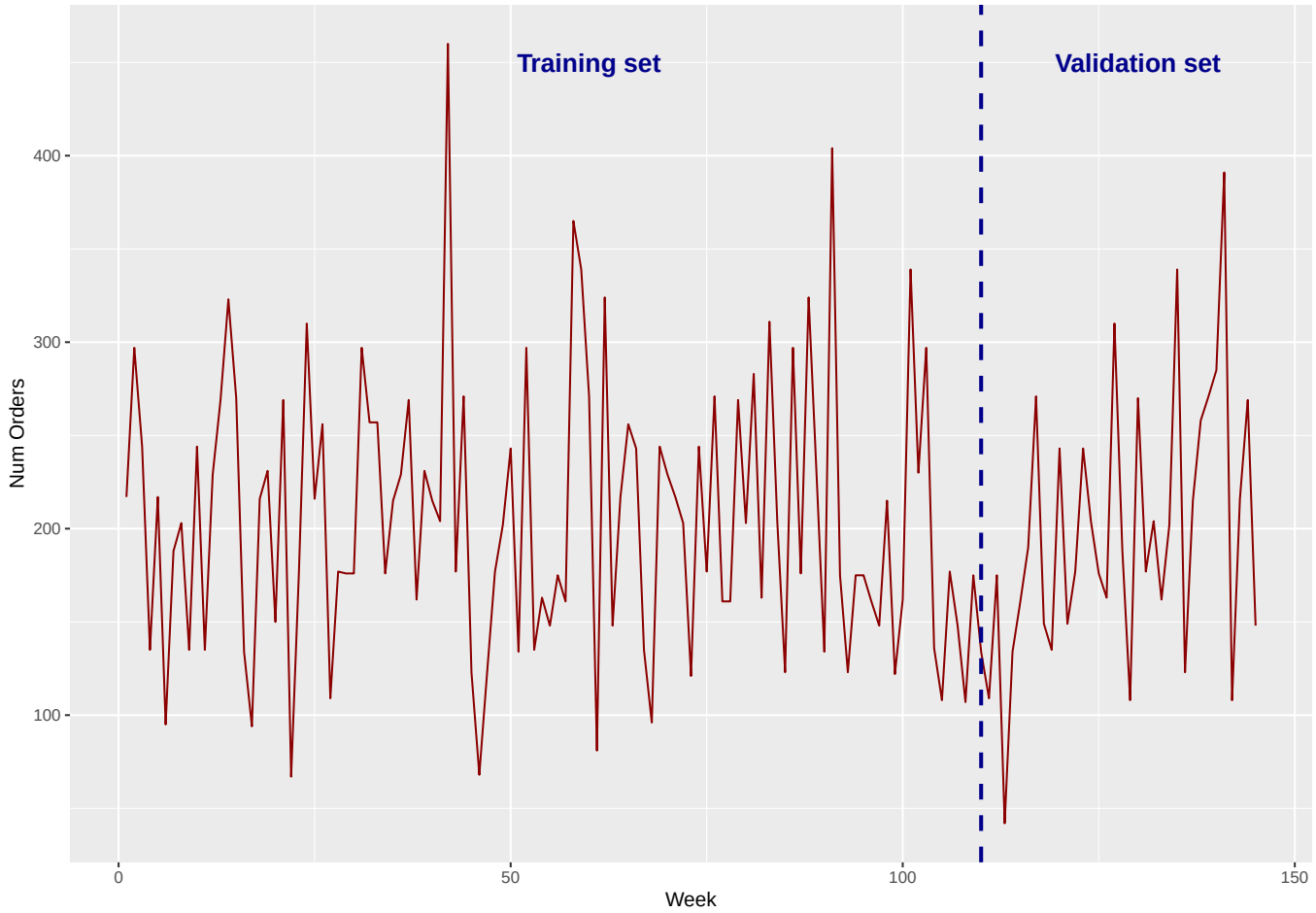
The model has $1 + 1 + 3926 + 3926 = 7854$ scalar parameters, that in the simple formulation there are 2 parameters for each model, so $2 \cdot 77 \cdot 51 = 7854$ total parameters.

Validation set and Test set

Dealing with time series data means that standard cross validation is not a viable option, since it break the temporal dependency.

We instead reserved a **validation set**, taking the **last set of observations**. The test set are the next 10 week, and the true number of orders stands on Kaggle.

Validation set and Test set

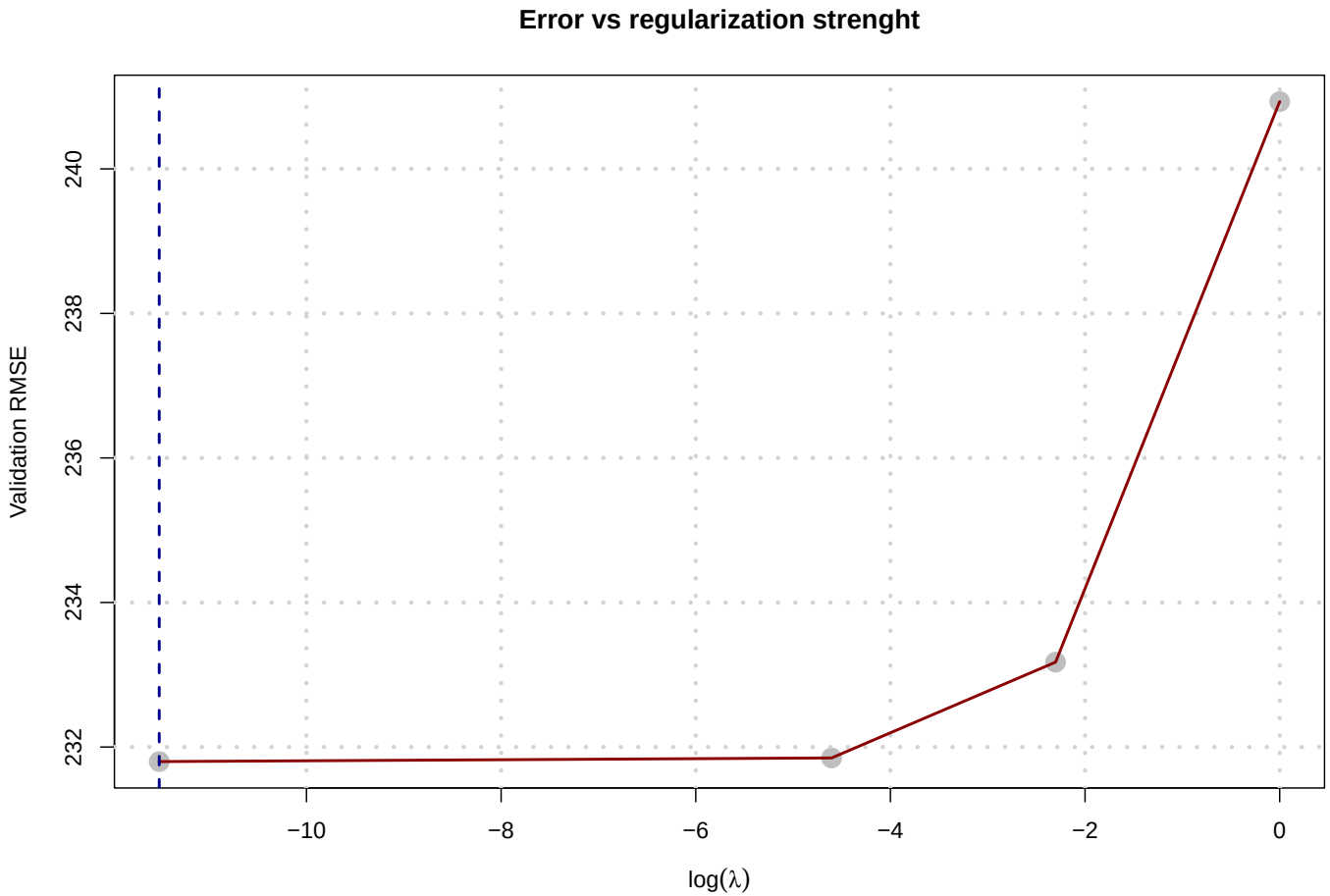


Regularization

We added **elastic-net regularization** in the estimation process:

$$\hat{\beta} = arg \min_{\beta \in \mathbb{R}^p} \left(\sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \sum_{j=1}^{p} \left(\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j| \right) \right)$$

No regularization is needed

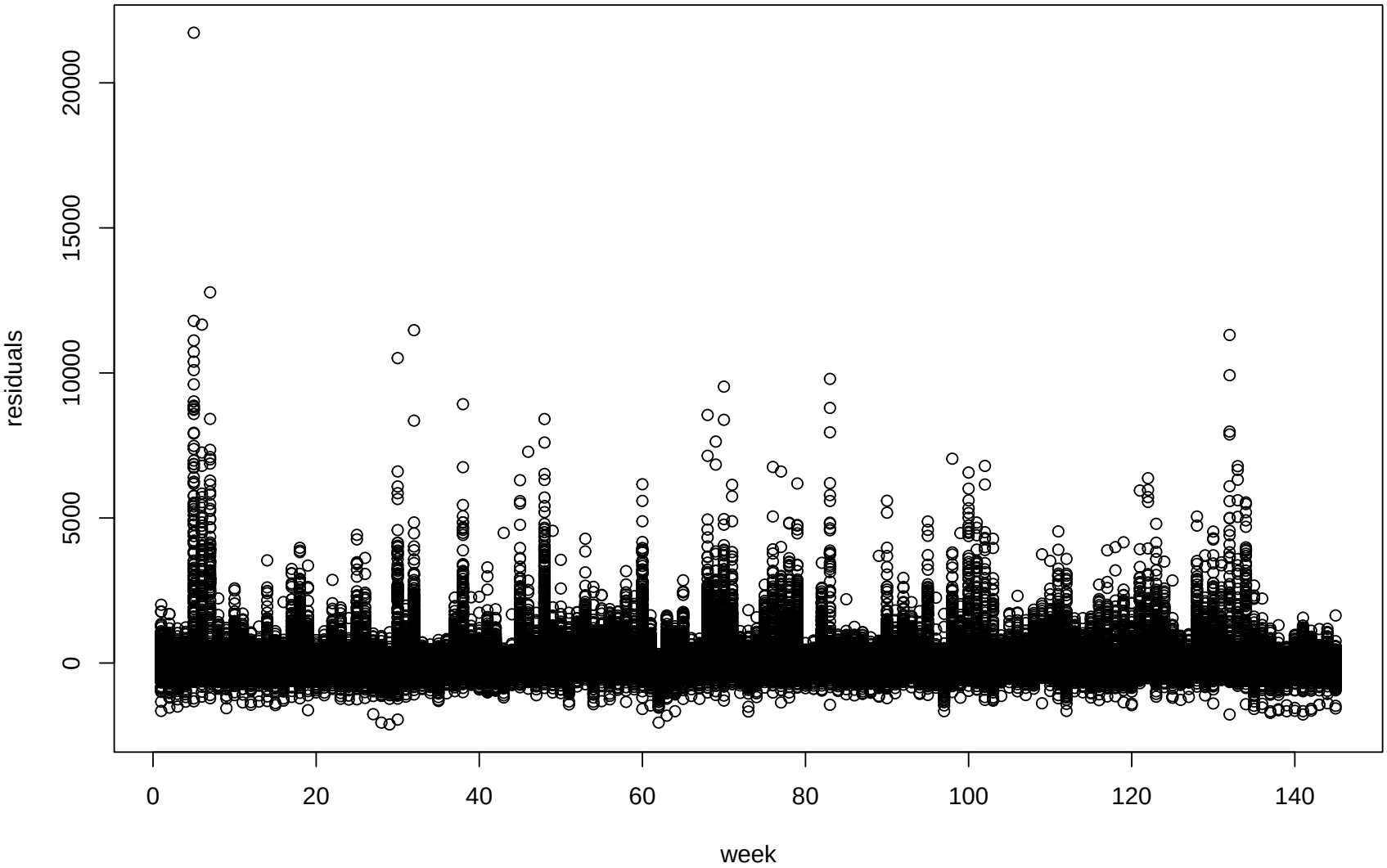


Results on validation set

Model	RMSE	MAE
Mean	350.9743	211.5107
LM	240.9311	106.4178
LM on ln(y)	367.1824	181.3026

```
##   meal_id center_id      id week checkout_price base_pri
## 1    1062         77 1278728   88          158.11     177
## 2    1062         89 1152882   23          160.05     182
## 3    1062        102 1090095   48          173.66     174
## 4    1062        177 1265569   63          178.51     177
## 5    1062         73 1354801   80          176.60     175
## 6    1062         91 1275390   89          163.99     184
##   emailer_for_promotion homepage_featured num_orders cit
## 1                     0                 0        324
```

Residuals of linear regression



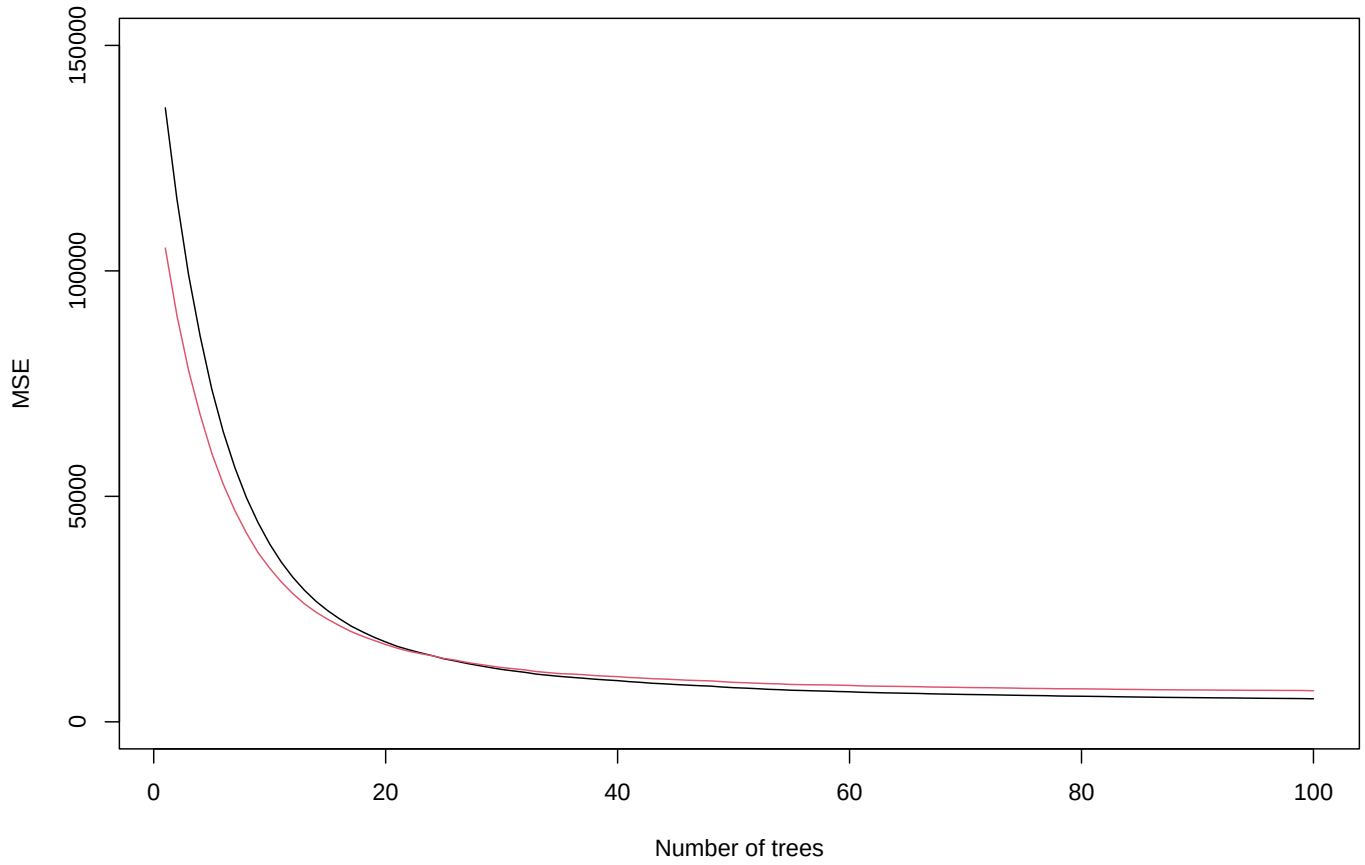
Gradient Boosting

We're going to apply GB to predict the proper **number of orders** using the residuals of the linear model as a new predictors.

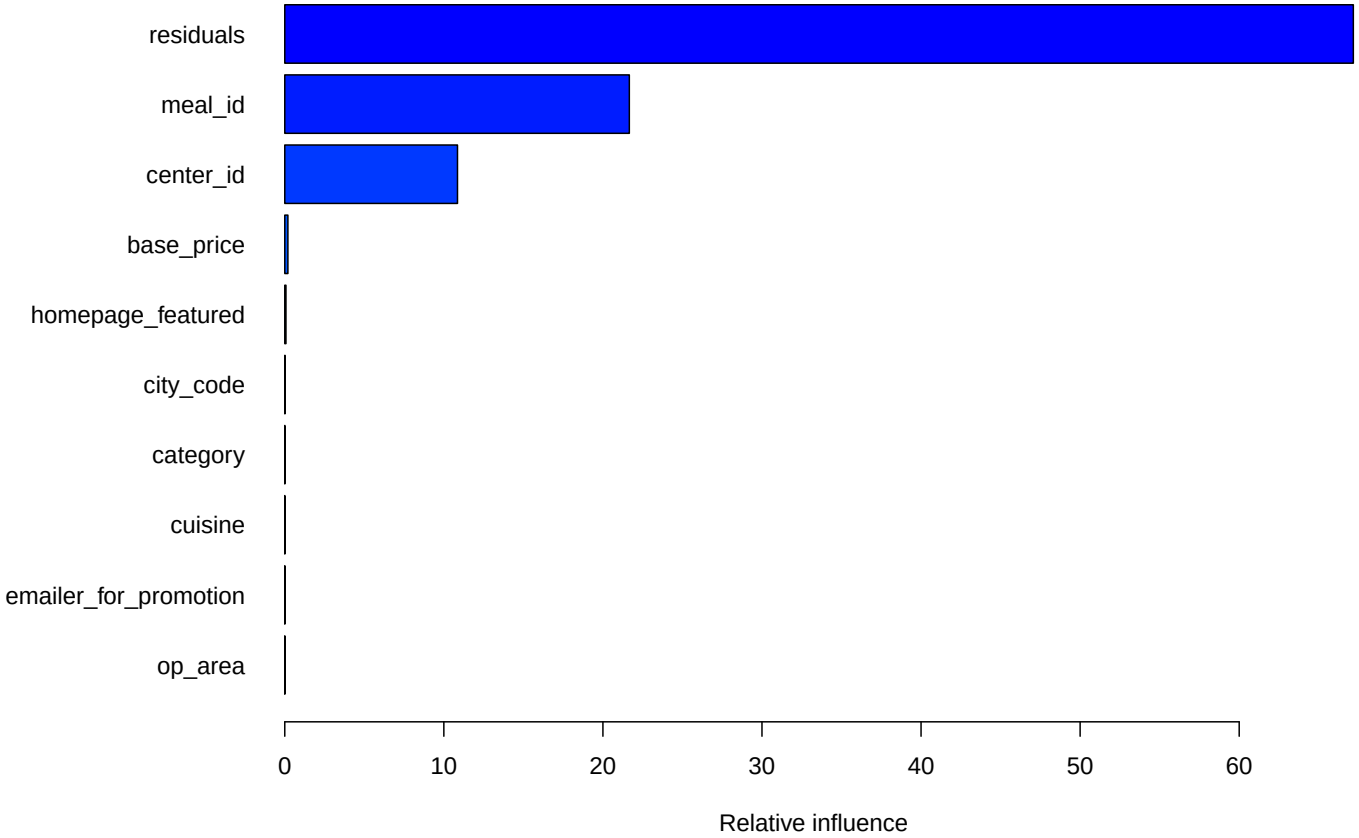
```
##  [1] "meal_id"                "center_id"                "ba
##  [4] "emailer_for_promotion" "homepage_featured"        "nu
##  [7] "city_code"              "region_code"              "ce
## [10] "op_area"                "category"                 "cu
## [13] "residuals"
```

```
## Loaded gbm 2.1.8
```

Training

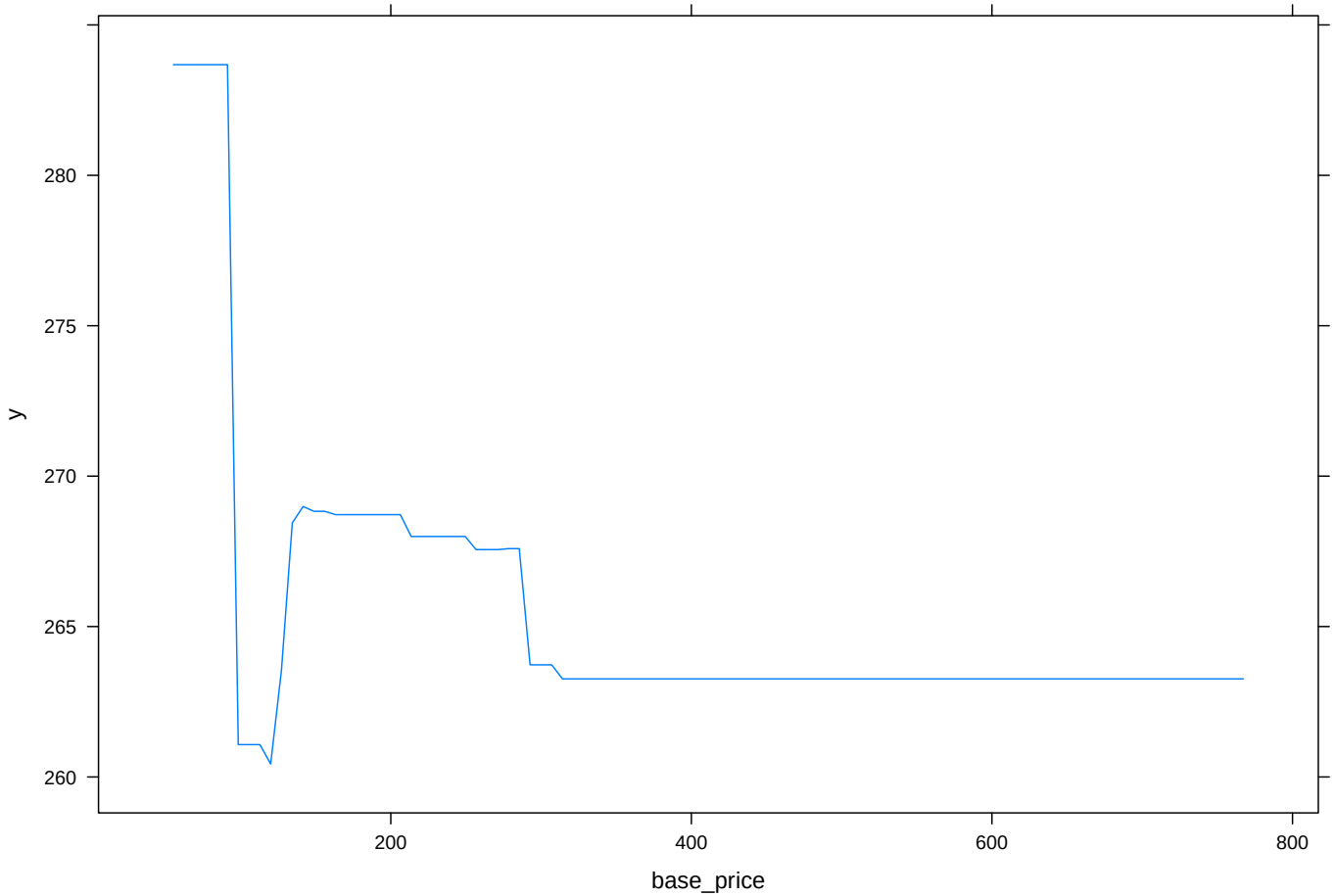


Relative influence plot

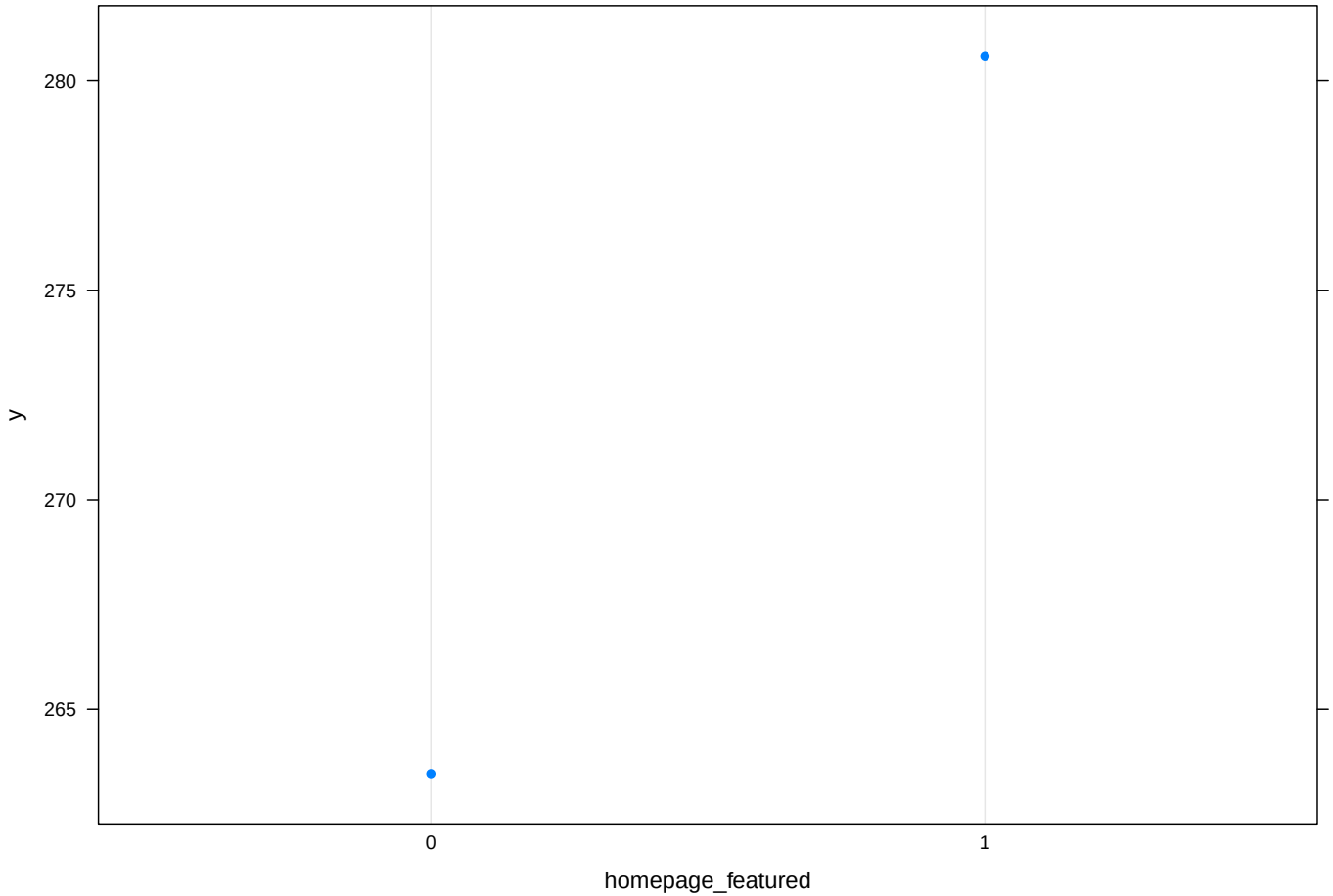



```
##                                  var      rel.inf
## residuals                  residuals 67.176496073
## meal_id                      meal_id 21.661007132
```

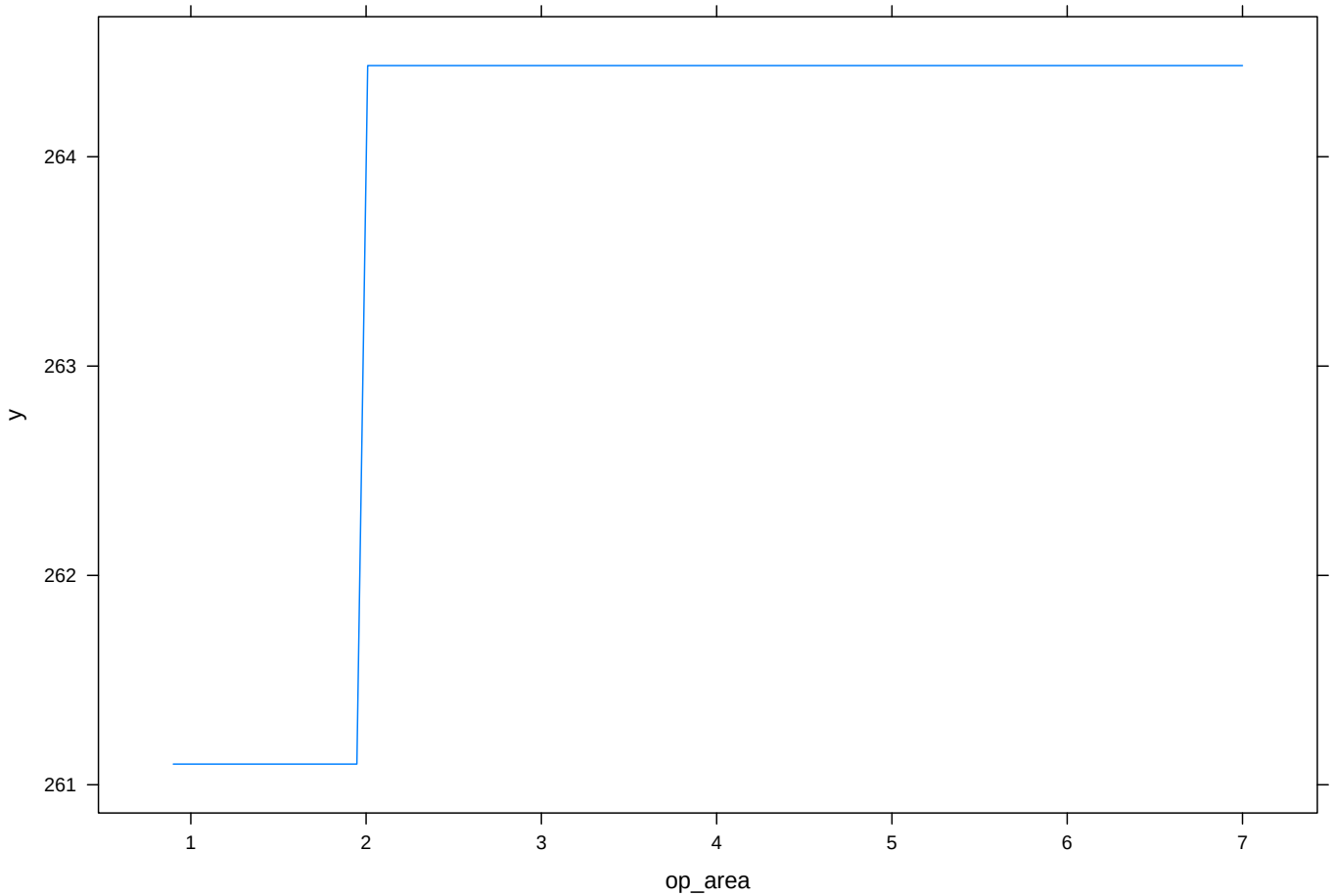
Marginal effects



Marginal effects



Marginal effects



Marginal effects