Data Analysis Project 1

MA8701

Group 5 : Yellow Submarine

15 February, 2021

Note on Open Science

To pursue the idea of reproducible research, the chosen dataset as well as the code for our analysis are publicly accessible:

- dataset: https://data.ub.uni-muenchen.de/2/1/miete03.asc
- code: https://github.com/FlorianBeiser/MA8701

The Data Set

In this project, we analyse a real dataset using shrinkage methods. For our project work we use the Munich Rent 2003 data set as described in https://rdrr.io/cran/LinRegInteractive/man/munichrent03.html. The data set has 12 covariates, of which many are suffering multicollinearity, a brief introduction to these parameters are listed below:

- `nmqm`: rent per square meter (double)
- `wfl`: area in square meters (int)
- `rooms`: number of rooms (int)
- `bj`: year of construction (factor)
- `bez`: district (factor)
- `wohngut`: quality of location (int)
- `wohnbest`: high quality of location (int)
- `ww0`: hot water supply available (int)
- `zh0`: central heating (int)
- `badkach0`: tiled bathroom (int)
- `badextra`: high-quality bathroom (int)
- `kueche`: upscale kitchen equipment (int) and the response
- `nm`: rental price (double).

For the data anaysis, the aim is to perform regression. Our data set is suited for that, since it suffers from multicolinearity as we see in Figure 1. For further data analysis, we store the data set in an R data frame.

Regression

We start with a vanilla LM regression for reference. Only significant coeffcients are printed. Clearly, the area `wfl` is strongly related to the rent price. Surprisingly in the regression, the significance of different `bj`s and `bez`s varies a lot.

```
##             summary.lm_mod..coef.summary.lm_mod..coef...4.....0.05..4..1.4.
## (Intercept)                                                     6.944363e-09
## wfl                                                            1.183420e-130
## rooms                                                           4.474346e-02
```

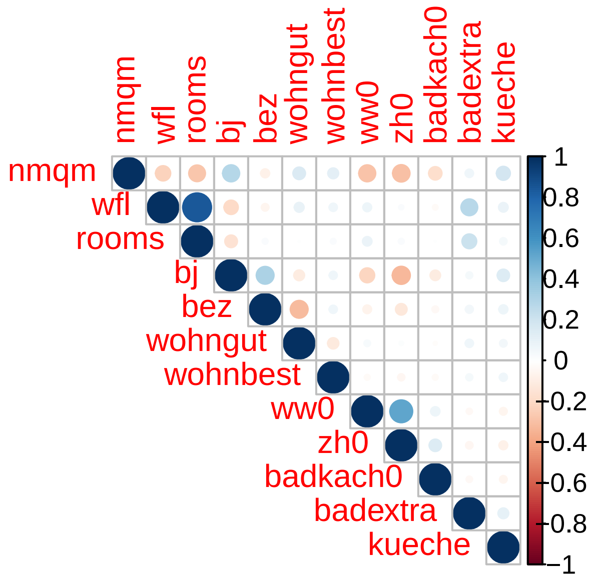

Figure 1: Correlation between the covariates

```
## bj1924                                                    3.936400e-07
```

Shrinkage

After we saw the results for the plain linear regression, we continue with the shrinkage methods. For comparison, ridge method and group lasso are both applied.

Ridge

```r
start <- glmnet(x = x_mod, y = y_mod, standardize = FALSE, alpha = 0)
autolambda <- start$lambda
newlambda <- c(autolambda, 10, 5, 3, 1, 0.5, 0.1)  # add more to approach zero lambda
ridge_fit <- glmnet(x_mod, y_mod, standardize = FALSE, alpha = 0, lambda = newlambda)
# plot(ridge_fit, xvar = 'lambda', label = T)
cv.ridge <- cv.glmnet(x_mod, y_mod, standardize = FALSE, alpha = 0, lambda = newlambda)
print(paste("The lamda giving the smallest CV error", cv.ridge$lambda.min))
```

```
## [1] "The lamda giving the smallest CV error 0.1"
```

```r
print(paste("The 1sd err method lambda", cv.ridge$lambda.1se))
```

```
## [1] "The 1sd err method lambda 10"
```

```r
plot(cv.ridge)
```

```r
plot(ridge_fit, xvar = "lambda", label = T) + abline(v = log(cv.ridge$lambda.1se))
```

```
## integer(0)
```

Lasso

For the λ with one standard deviation, we observe that many of the `bj`s and `bez`s get shrinked, but not all of them - and the values differ from the linear regression. Whereas the other kept covariants roughly keep their parameter.

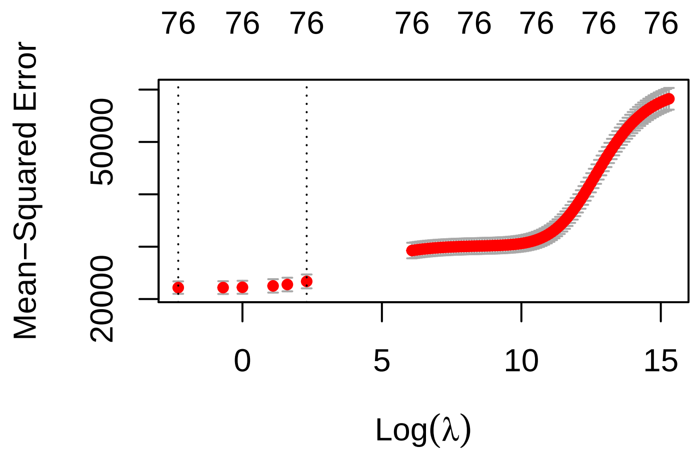

Figure 2: Coefficient path along lambda variation

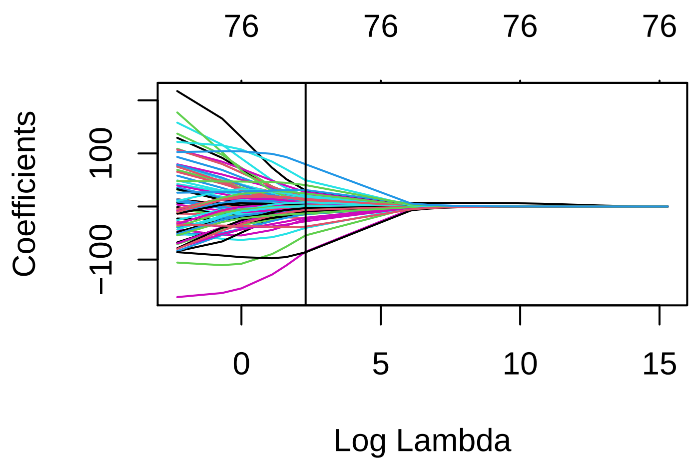

Figure 3: Coefficient path along lambda variation

Above we considered a fixed λ, now we analyse which λ is optimal using cross validation.

In the grouped lasso, the `bj` and `bez` are all shrinked or are all inclueded, respectively. This coincides better with our intuition, that this criterion is considered or not considered. Whereas in the regression and lasso before, just some years of construction and some areas where significant.