

# The Merit Order Effect in the German Power Market

Felix Germaine

Manuel Pfeuffer

Bruno Puri

Ladislaus von Bortkiewicz Chair of Statistics

Humboldt-Universität zu Berlin

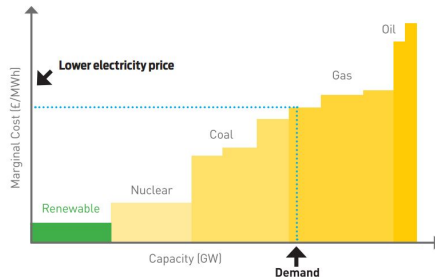
Statistical Programming Languages

<https://github.com/mpff/spl2018-bfm>



## Motivation

We want to measure the merit order effect for the German power market, based on Clò, Cataldi, and Zoppoli 2015.



## Regression

Regression equation:

$$\text{price} = \beta_0 + \beta_1 \cdot \text{dem} + \beta_2 \cdot \text{wind} + \beta_3 \cdot \text{solar} + \gamma \cdot \text{ymd} + \epsilon \quad (1)$$

**price:** Mean daily day ahead price

**dem:** Forecasted daily demand

**solar:** Forecasted daily electricity production from solar

**wind:** Forecasted daily electricity production from wind

**ymd:** Year, month, day dummies



## Data Sources

- price:** Day Ahead Price, since 2011 (Elsport Prices Data).
- dem:** Day Ahead Demand, since 2015, DE+AT+LUX (entsoe)
- solar:** Day Ahead Forecast, since 2012, DE (netztransparenz.de)  
Day Ahead Forecast, since 2015, AT (apg.at)
- wind:** Day Ahead Forecast, since 2012, DE (netztransparenz.de)  
Day Ahead Forecast, since 2015, AT (apg.at)

### Probleme:

1. Different timeframes and -frequencies.
2. NA's a problem for daily averages and sums.
3. LUX in demand a (big) problem?
4. Day Ahead Price the same for DE and AT?



## Data Cleaning

```
df.pun.0 <- read.csv("source/Elspot_xxx.csv")
df.pun <- subset(df.pun.0, select = c(HourUTC,
SpotPriceEUR))
colnames(df.pun) = c("TIME", "PUN")
df.pun$TIME <- ymd_hm(df.pun$TIME)
df.pun <- time.FRAME(df.pun)
df.pun <- aggregate(list("PUN" = df.pun$PUN),
list("TIME" = cut(df.pun$TIME, "1 day")), FUN = mean)
...
# Bind final Dataframe
df <- cbind(df.pun, df.dm, df.solar, df.wind)
```



## Exploratory Analysis

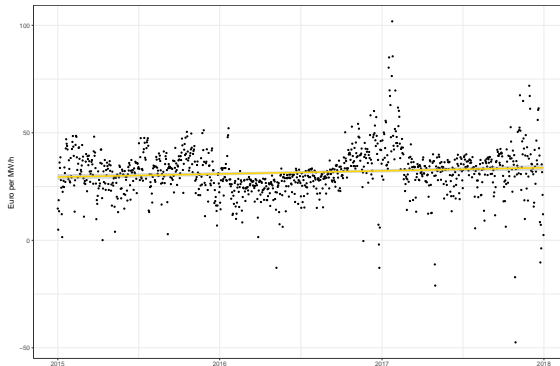


Figure 1: Day-Ahead Price



## Exploratory Analysis

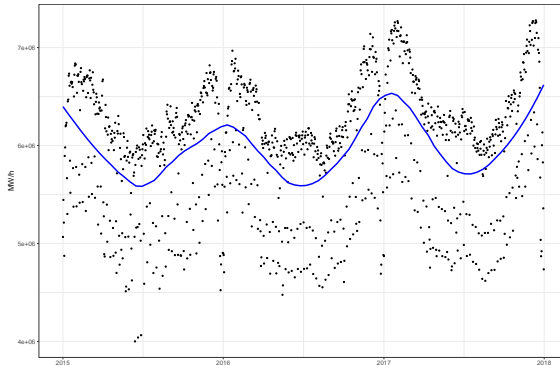


Figure 2: Day-Ahead Demand



## Exploratory Analysis

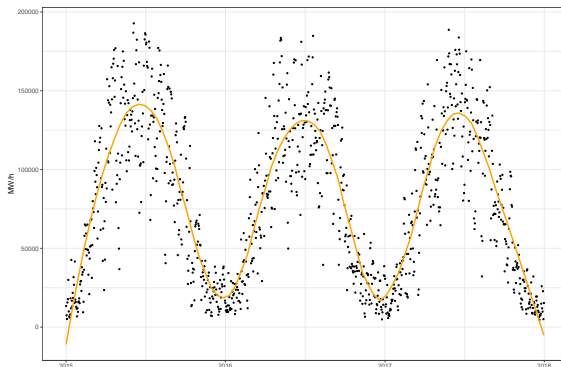


Figure 3: Day-Ahead Solar Production





## Exploratory Analysis

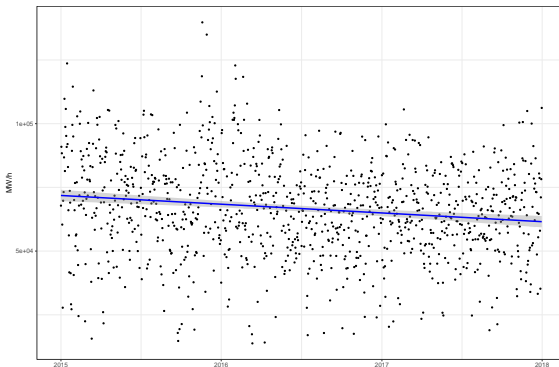


Figure 4: Day-Ahead Wind Production



## Exploratory Analysis

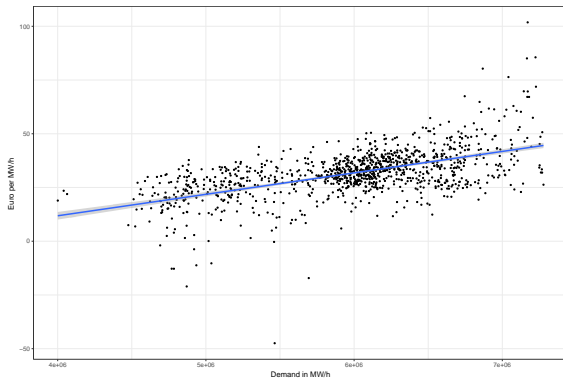


Figure 5: Price on Demand



## Exploratory Analysis

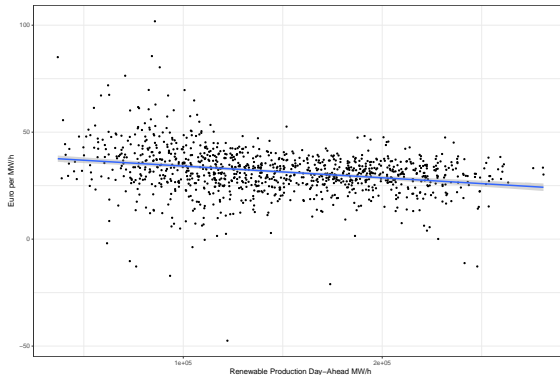


Figure 6: Day-Ahead Price on Renewables Production



## Problems with Missing Values [in Arbeit]



## Interpolation [in Arbeit]

Function for removing missing values

**Example function:**  $x[i] \leftarrow \text{mean}(x[(i-1):(i+1)])$

- Input - numerical vector and index of missing values
- Computes mean of values next to missing value
- Output - numerical vector without missing values

Alternative: Use time "based" interpolation

- Is missing values problem big?
- Otherwise keep missing Values?



## Time-Dummy generating function:

|            | PUN    | DEMAND.DAY-AHEAD.MW/h | SOLAR.MW/h | WIND.MW/h |
|------------|--------|-----------------------|------------|-----------|
| 2015-01-01 | 354.32 | 5066593               | 13412.739  | 67844.38  |
| 2015-01-02 | 118.90 | 5444787               | 7497.092   | 90930.91  |
| 2015-01-03 | 447.53 | 5213461               | 5199.207   | 88375.82  |
| 2015-01-04 | 324.56 | 4874273               | 15651.090  | 60406.97  |
| 2015-01-05 | 868.25 | 5894292               | 17744.375  | 27798.76  |
| 2015-01-06 | 808.05 | 5980533               | 29969.718  | 69774.96  |

```
#Step1: Create the dummy variables for years, months and days
## 1.a Dummy variables for Years
```

```
Year.Dummy.matrix <- matrix(nrow = length(FullDat.xts[,1]), ncol=Year.max.number-Year.min.number+1)
colnames(Year.Dummy.matrix) = Year.Vector

for (i in 1:length(Year.Vector)) {
  Year.Dummy.matrix[,i] <- format(index(FullDat.xts), "%Y")== Year.Vector[i]
}
```

- Creates a dummy matrix for: year, month, day o.t. week.
- Function adapts the amount of years directly from the data.
- y/m/d to be omitted, bc. of the dummy variable trap, can be specified.



## Regression and Tests

$$\text{price} = \beta_0 + \beta_1 \cdot \text{dem} + \beta_2 \cdot \text{wind} + \beta_3 \cdot \text{solar} + \gamma \cdot \text{ymd} + \epsilon$$

**Step 1:** Check for stationarity of the variables (augmented dickey fuller Test) → All variables stationary except solar.

**Step 2:** Perform the OLS regression.

**Step 3:** Check for Heteroscedasticity and Autocorrelation.  
→ Breusch-Pagan Test for heteroscedasticity.  
→ Durbin-Watson test for autocorrelation.  
→ Plot the autocorrelation structure.

**Step 4:** If okay [AR(1)], use the Prais-Winsten generalized estimation method.

**Step 5:** Check the new autocorrelation structure.



## Regression and Tests

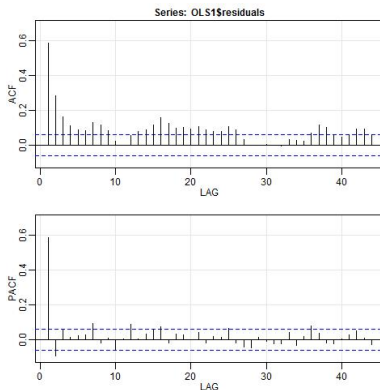


Figure 7: **OSL, ACF and PACF** Autocorrelation = TRUE (durbin-watson)





## Regression and Tests

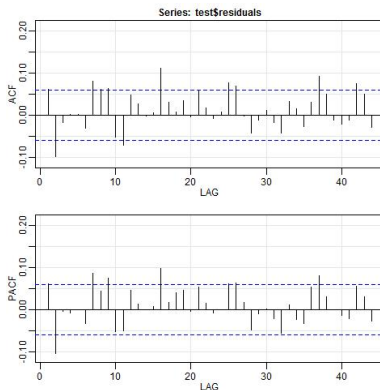


Figure 8: **Prais-Winsten, ACF and PACF**



## Regression and Tests

$$\text{price} = \beta_0 + \beta_1 \cdot \text{dem} + \beta_2 \cdot \text{wind} + \beta_3 \cdot \text{solar} + \gamma \cdot \text{ymd} + \epsilon$$

```

coefficients:
      Estimate Std. Error t value Pr(>|t|)
Intercept -3.541e+00  5.205e+00  -0.680  0.496445
DEM        6.751e-06  8.914e-07   7.573  7.82e-14 ***
SOLAR      -9.209e-06  9.370e-06  -0.983  0.325906
WIND       -7.679e-05  1.029e-05  -7.465  1.72e-13 ***
X2016      -2.862e+00  1.352e+00  -2.116  0.034558 *
X2017       6.009e-01  1.377e+00   0.436  0.662736
X02        -4.393e+00  2.302e+00  -1.908  0.056604 .
X03        -6.457e+00  2.449e+00  -2.637  0.008495 **
X04        -3.994e+00  2.605e+00  -1.533  0.125561
X05        -5.292e+00  2.656e+00  -1.992  0.046614 *
X06        -2.150e+00  2.692e+00  -0.798  0.424759
X07        -1.322e-01  2.652e+00  -0.050  0.960263
X08        -1.724e+00  2.648e+00  -0.651  0.515092
X09        -1.009e+00  2.542e+00  -0.397  0.691371
X10         1.165e+00  2.442e+00   0.477  0.633527
X11        -4.452e-01  2.420e+00  -0.184  0.854089
X12        -3.498e+00  2.321e+00  -1.507  0.131985
Monday     4.745e+00  1.129e+00  4.203  2.86e-05 ***
Tuesday    5.336e+00  1.350e+00  3.952  8.25e-05 ***
Wednesday  4.771e+00  1.402e+00  3.404  0.000689 ***
Thursday   4.210e+00  1.392e+00  3.025  0.002543 **
Friday     3.994e+00  1.303e+00  3.064  0.002237 **
Saturday   2.513e+00  6.881e-01  3.652  0.000273 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.595 on 1073 degrees of freedom
Multiple R-squared:  0.7858,    Adjusted R-squared:  0.7812
F-statistic: 171.1 on 23 and 1073 DF,  p-value: < 2.2e-16

```

Figure 9: Prais-Winsten



## Our Questions [in Arbeit]

How to handle missing values correctly?

What does 'testing our program' mean?

