Day 1: Introduction

We talked today about the massive transformation that electricity markets are witnessing, with the rapid growth of renewable power and explicity goal of fully decarbonizing the electricity market in coming years.

In this practice session, we will examine time series data from the Spanish electricity market, which has substantial intermittent renewable power (wind and solar).

The data have been collected from publicly available sources (Red Eléctrica de España and OMIE, among others). The data are from the paper "Measuring the Impact of Wind Power in the Spanish Electricity Market," by Claire Petersen, Mar Reguant, and Lola Segura.

We need to load packages in Julia, similar to the import function in Python or the library functionality in R. *Pluto* will install the libraries automatically, but in Julia we need to **install the libraries** as follows:

```
using Pkg
Pkg.add("LibraryName")
```

To **load the libraries**, we use the command `using`.

Here we will be loading a bunch of libraries so that we can load and use the data (`DataFrames`, `CSV`), compute statistics and manipulate data (`Statistics`) and make some nice plots (`Plots`, `Binscatters`). We will also be running some fixed-effects regressions (`FixedEffectModels`).

```
begin
    using DataFrames
    using Statistics
    using CSV
    using Plots
    using Binscatters
    using FixedEffectModels
end
```

We load the data using the CSV syntax (`CSV.read`) into a data frame called `df`. `first(df,5)` gives us a snapshot of the data.

	year	month	day	hour	dayofweek	demand	demand_forecast	wind	w
1	2009	5	1	1	5	25.337	25.021	5.7625	5
2	2009	5	1	2	5	23.478	23.044	5.7461	5
3	2009	5	1	3	5	21.859	21.685	5.786	5
4	2009	5	1	4	5	20.931	20.408	5.9837	5
5	2009	5	1	5	5	20.371	19.586	6.0831	5
6	2009	5	1	6	5	19.809	18.871	6.1295	5
7	2009	5	1	7	5	19.681	18.383	6.1246	5
8	2009	5	1	8	5	19.132	17.965	6.2554	5
9	2009	5	1	9	5	19.688	18.178	6.1949	6
10	2009	5	1	10	5	21.342	19.59	5.9185	6
more									
78731	2018	6	1	24	5	26.434	26.597	3.6417	3

```
begin
    df = CSV.read("data_spain.csv", DataFrame)
    df
end
```

Summary Statistics

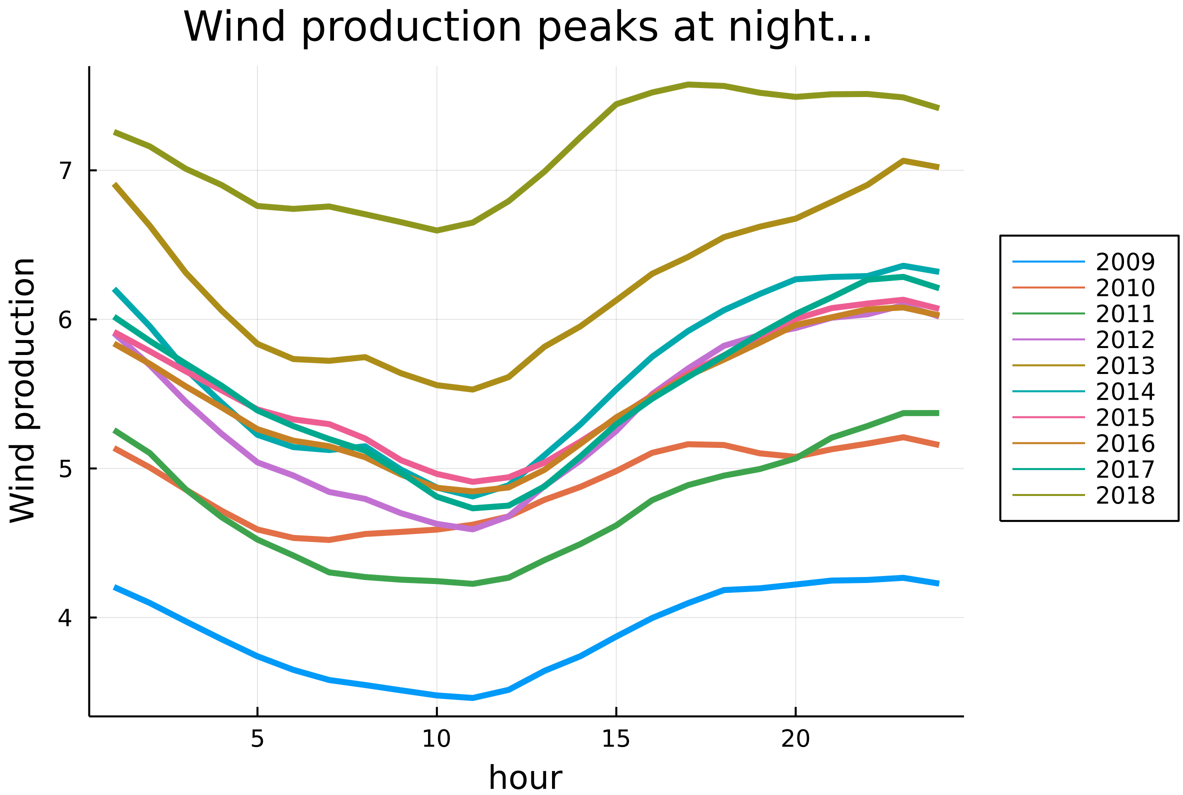
We start by displaying some statistics and plot hourly and yearly patterns of wind production and electricity demand.

Variables in Julia are defined by colons (:). eltype determines the type of the variable. Note that Julia differentiates between Integers (1) and Floating-Point (1.0)

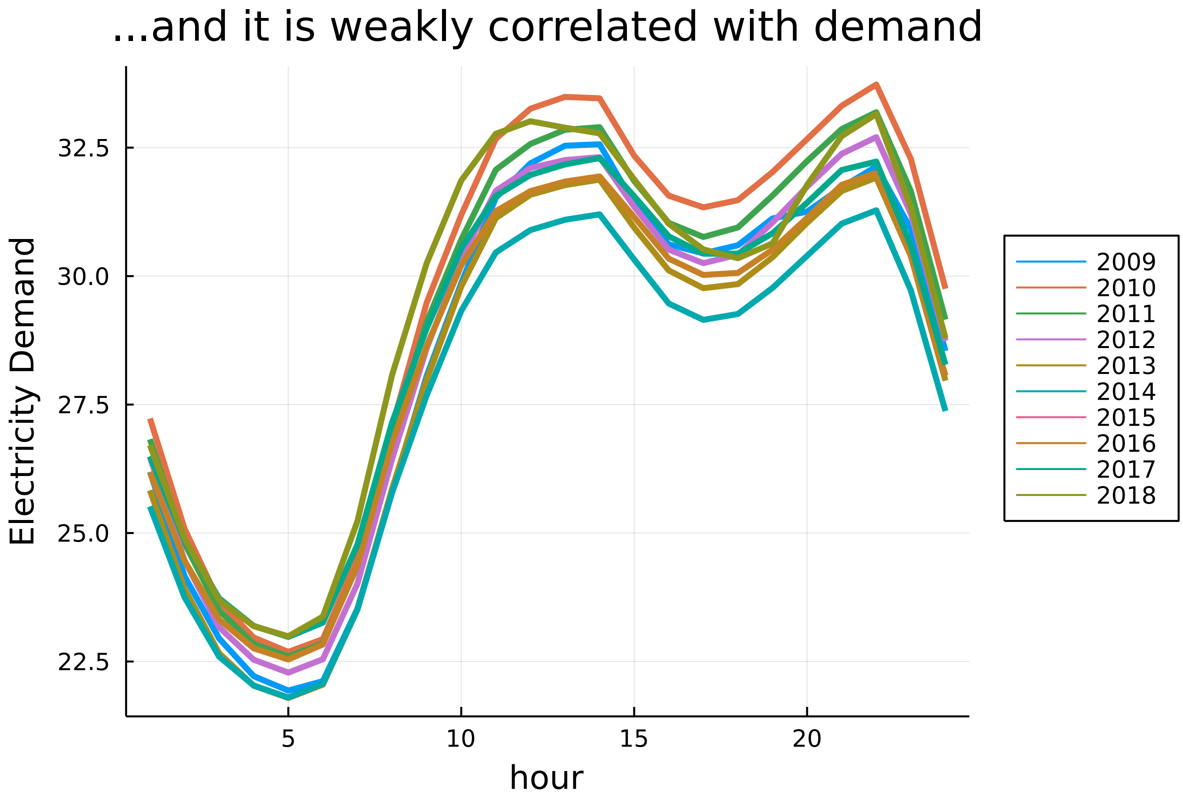
	variable	mean	min	median	max	nmissing	eltype
1	:year	2013.39	2009	2013.0	2018	0	Int64
2	:month	6.54388	1	7.0	12	0	Int64
3	:day	15.7109	1	16.0	31	0	Int64
4	:hour	12.4865	1	12.0	24	0	Int64
5	:dayofweek	3.00334	0	3.0	6	0	Int64
6	:demand	28.6556	17.096	28.827	43.588	24	Union{Missing, Flo
7	:demand_forecast	28.6898	14.38	28.853	43.568	0	Float64
8	:wind	5.38397	0.1552	4.7899	17.4963	0	Float64
9	:wind_forecast	5.29389	0.398	4.698	16.768	1	Union{Missing, Flo
10	:wholesale_price	44.8322	0.0	46.5	145.0	0	Float64
11	:system_costs	3.96826	-1.81	3.22	99.39	0	Float64
12	:emis_tCO2	7087.11	0.0	7204.52	15772.7	1	Union{Missing, Flo

```
describe(df)
```

In order to plot hourly and yearly patterns, we first need to combine the data at those levels. For that, we first define the groups for which the functions will be applied using `groupby`. `combine` is then used to compute the specified summary statistic. Finally, we rename the variable as `wind_mean`.




```
begin
    #compute the mean for each hour and year
df_mean = combine(groupby(df, ["hour","year"]), :wind => mean => :wind_mean,
    :demand => mean => :demand_mean);

plot(df_mean.hour, df_mean.wind_mean, group = df_mean.year,
    seriestype = :line, linewidth = 3,
    title = "Wind production peaks at night...",
    xlabel = "hour",
    ylabel = "Wind production",
    legend = :outerright)

end
```



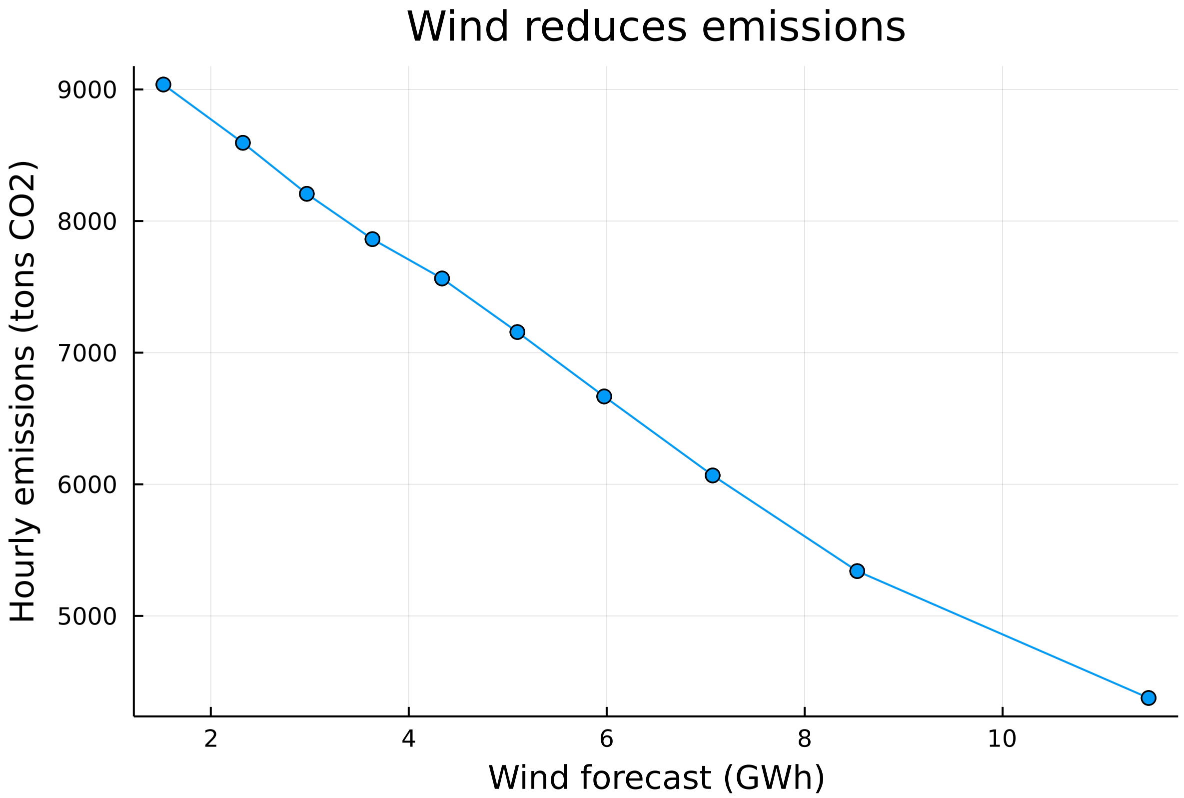

```
begin
plot(df_mean.hour, df_mean.demand_mean, group = df_mean.year,
    seriestype = :line, linewidth = 3,
    title = "...and it is weakly correlated with demand",
    xlabel = "hour",
    ylabel = "Electricity Demand",
    legend = :outerright)

end
```

The impacts of wind: a visual exploration

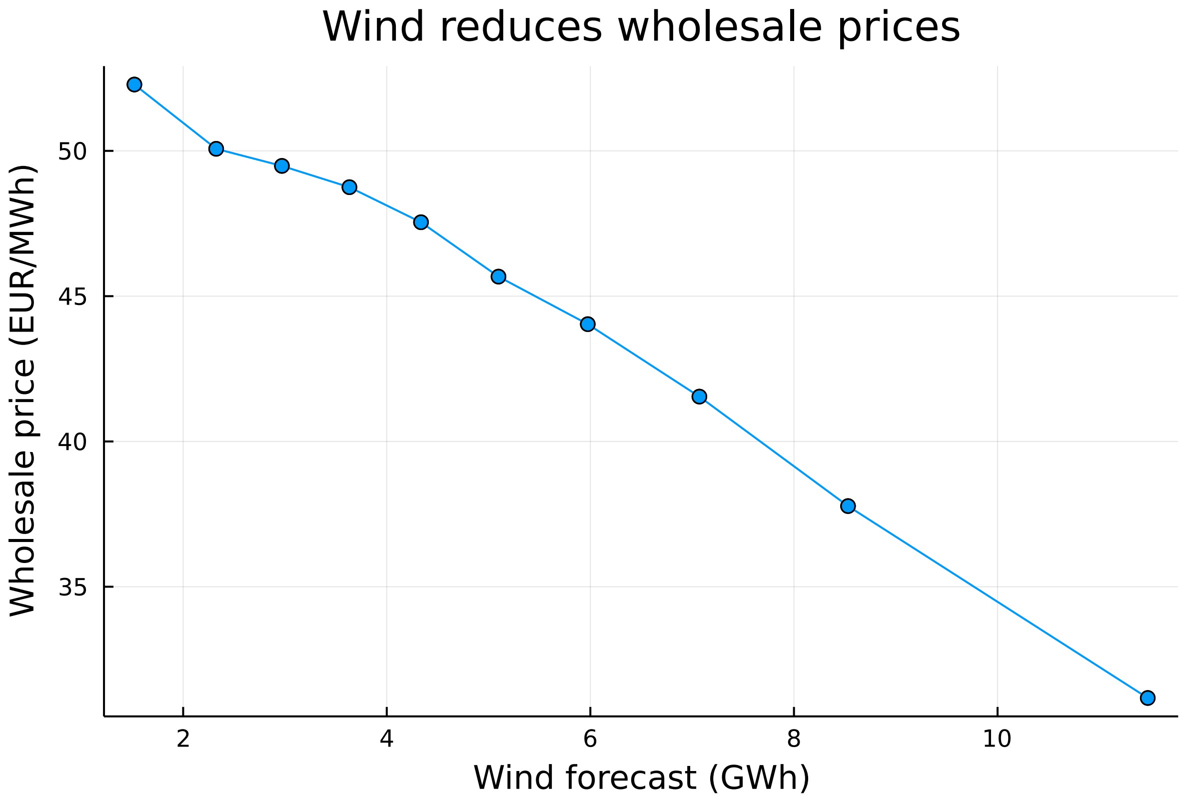
We will be plotting the **impacts of wind** on several outcomes of interest:

- Emissions
- Wholesale prices
- System costs
- Wholesale prices + system costs

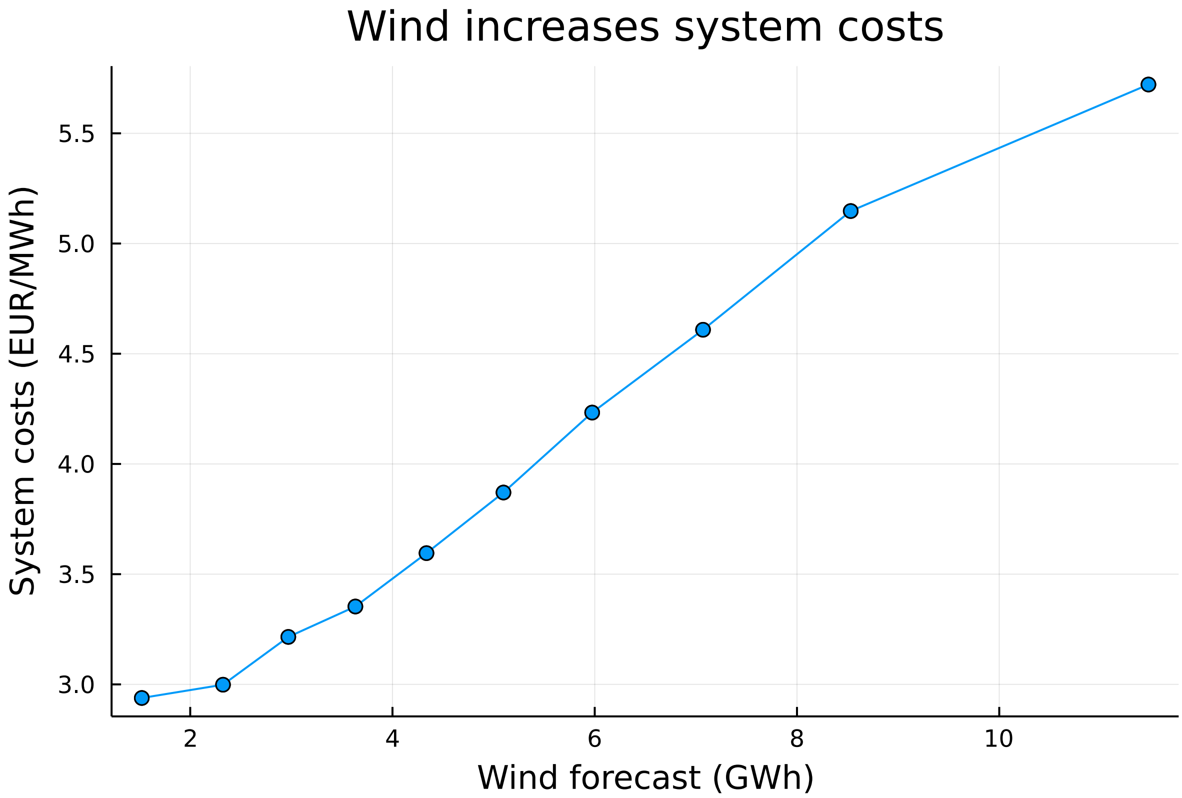
We will be using the library `Binscatters` for plotting.



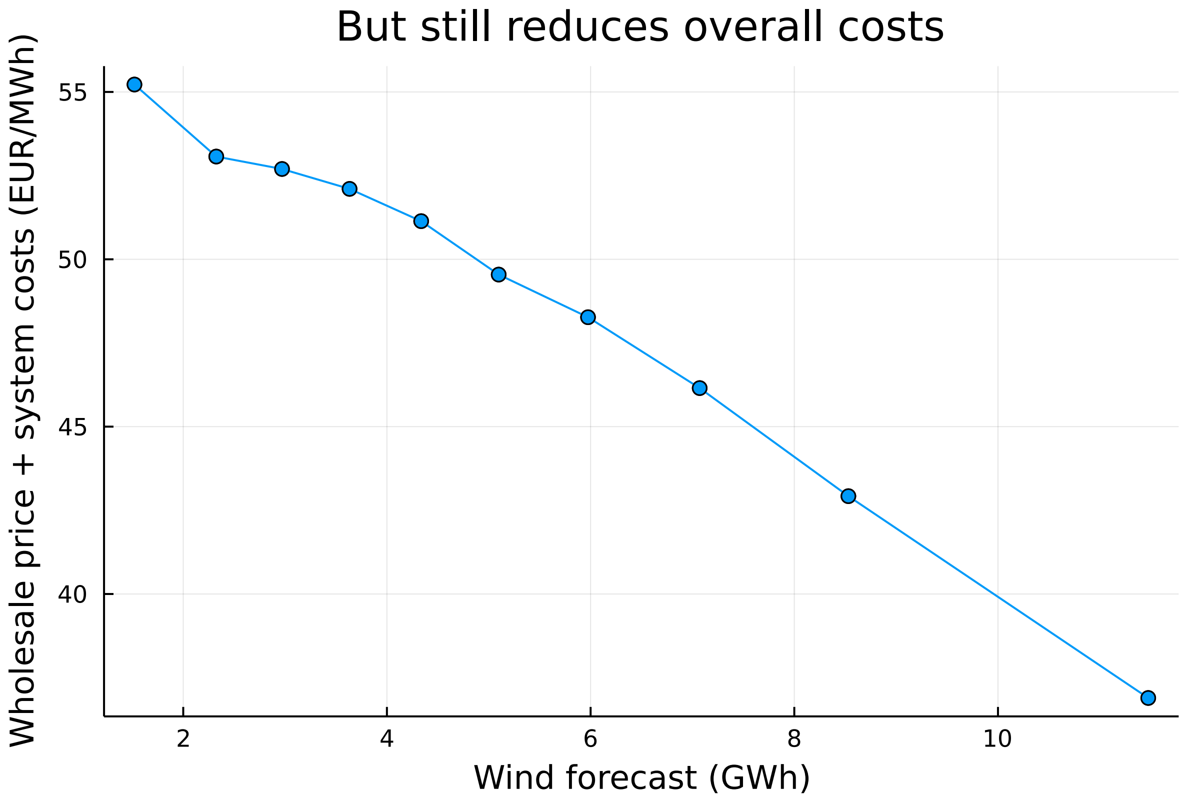

```
begin
binscatter(df, @formula(emis_tCO2 ~ wind_forecast), 10,
        seriestype = :scatterpath,
        title = "Wind reduces emissions",
        xlabel = "Wind forecast (GWh)",
        ylabel = "Hourly emissions (tons CO2)")
#we can add new specifications
#binscatter!(df, @formula(emis_tCO2 ~ wind), 10,seriestype = :scatterpath)
end
```




```julia
begin
binscatter(df, @formula(wholesale_price ~ wind_forecast), 10,
        seriestype = :scatterpath,
        title = "Wind reduces wholesale prices",
        xlabel = "Wind forecast (GWh)",
        ylabel = "Wholesale price (EUR/MWh)")
#we can add controls
#binscatter!(df, @formula(wholesale_price ~ wind_forecast + demand_forecast + fe(year)             + fe(month) + fe(hour)), 10,seriestype = :scatterpath)
end
```




```
binscatter(df, @formula(system_costs ~ wind_forecast), 10,
        seriestype = :scatterpath,
        title = "Wind increases system costs",
        xlabel = "Wind forecast (GWh)",
        ylabel = "System costs (EUR/MWh)")
```

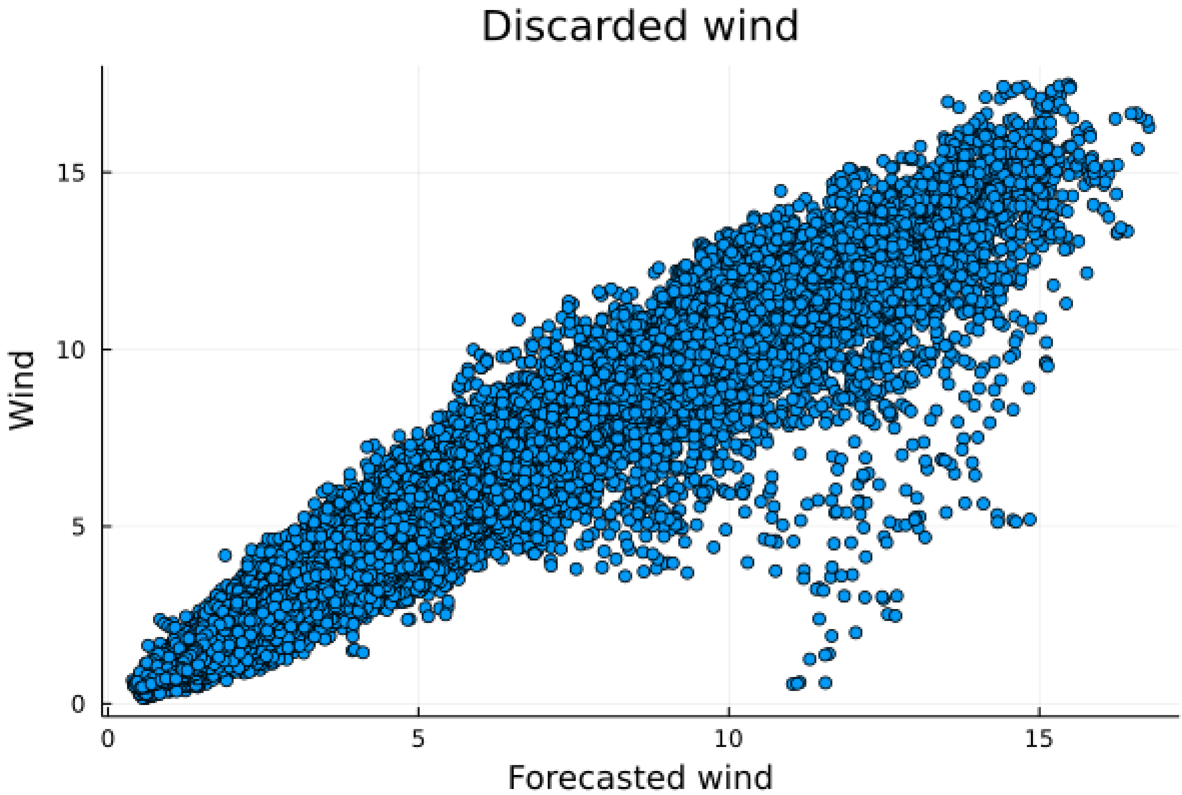



```
begin
    df.total_price = df.wholesale_price + df.system_costs
    binscatter(df, @formula(total_price ~ wind_forecast), 10,
        seriestype = :scatterpath,
        title = "But still reduces overall costs",
        xlabel = "Wind forecast (GWh)",
        ylabel = "Wholesale price + system costs (EUR/MWh)")
end
```

Wind endogeneity

One can estimate the effects of wind using a regression framework. However, it is important to keep in mind that wind production can be endogenous.

In moments of very high forecasted wind, it is often the case that wind is discarded. This can create an endogeneity problem.




```
scatter(df.wind_forecast, df.wind, xlabel="Forecasted wind", ylabel="Wind",
legend=false, title="Discarded wind")
```

We can examine the endogeneity problem in the context of assessing the impact of wind on reliability and other congestion costs ("system costs").

On days of very high wind, measured wind production could be lower than expected, leading to a downward bias in our estimates: a difficult day with lots of wind appears as a day with low levels of wind in the data.

To address this issue, one can use forecasted wind as an exogenous variable.

We will be running these regressions using the `FixedEffectModels` library.

```
reg_w =                         Fixed Effect Model
        ========================================================================
        Number of obs:              78731   Degrees of freedom:              23
        R2:                         0.228   R2 Adjusted:                  0.228
        F-Stat:                   2762.59   p-value:                      0.000
        R2 within:                  0.034   Iterations:                       6
        ========================================================================
        system_costs | Estimate  Std.Error t value Pr(>|t|) Lower 95% Upper 95%
        ------------------------------------------------------------------------
        wind         | 0.187218 0.00356196 52.5603    0.000  0.180236  0.194199
        ========================================================================
```

```
reg_w = reg(df, @formula(system_costs ~ wind + fe(year) + fe(month)))
```

```
Demean Variables: [================>                ]  1/2 [?25h [2K
```

```
reg_wf =                              Fixed Effect Model
         ========================================================================
         Number of obs:                  78730  Degrees of freedom:            23
         R2:                             0.235  R2 Adjusted:                0.235
         F-Stat:                       3562.29  p-value:                    0.000
         R2 within:                      0.043  Iterations:                     6
         ========================================================================
         system_costs   | Estimate  Std.Error t value Pr(>|t|) Lower 95% Upper 95%
         ------------------------------------------------------------------------
         wind_forecast | 0.224957 0.00376907 59.6849    0.000  0.217569  0.232344
         ========================================================================
```

```
reg_wf = reg(df, @formula(system_costs ~  wind_forecast + fe(year) + fe(month)))
```

Another possible problem is that system costs from wind production may be realized in hours with no wind. In this case, the hourly regression coefficient will be downward biased. To circumvent this issue, we can estimate the same regression at a daily level.

For that, we compute the total system costs as well as total wind power.

	day_id	year	month	wind_forecast	system_costs
1	"200951"	2009	5	147.466	52.62
2	"200952"	2009	5	116.489	41.43
3	"200953"	2009	5	122.897	39.33
4	"200954"	2009	5	148.368	62.05
5	"200955"	2009	5	121.773	57.24
6	"200956"	2009	5	50.878	57.12
7	"200957"	2009	5	48.955	36.98
8	"200958"	2009	5	63.604	42.66
9	"200959"	2009	5	71.073	32.17
10	"2009510"	2009	5	64.523	21.69
more					
3294	"201861"	2018	6	52.531	51.3

```
begin
df.day_id = string.(df.year,df.month,df.day)
#In Julia, row-wise operations are defined with a dot.

df_day = combine(groupby(df, ["day_id","year","month"]), :wind_forecast => sum =>
:wind_forecast, :system_costs => sum => :system_costs);

end
```

```
reg_d =                        Fixed Effect Model
        ========================================================================
        Number of obs:                  3293   Degrees of freedom:            23
        R2:                            0.507   R2 Adjusted:                0.504
        F-Stat:                      537.859   p-value:                    0.000
        R2 within:                     0.141   Iterations:                     6
        ========================================================================
        system_costs  | Estimate Std.Error t value Pr(>|t|) Lower 95% Upper 95%
        ------------------------------------------------------------------------
        wind_forecast | 0.255654 0.0110235 23.1918    0.000  0.234041  0.277268
        ========================================================================
```

```
reg_d = reg(df_day, @formula(system_costs ~  wind_forecast + fe(year) + fe(month)))
```

We can display the output of our regressions using the `RegressionTables` package (similar to `stargazer` in R).

```
begin
    using RegressionTables

    regtable(reg_w, reg_wf,reg_d, regression_statistics = [:nobs,:adjr2])

end
```

```
-----------------------------------------------
                           system_costs
                ------------------------------
                     (1)        (2)        (3)
-----------------------------------------------
wind            0.187***
                 (0.004)
wind_forecast              0.225***   0.256***
                            (0.004)    (0.011)
-----------------------------------------------
year                 Yes        Yes        Yes
month                Yes        Yes        Yes
-----------------------------------------------
Estimator            OLS        OLS        OLS
-----------------------------------------------
N                 78,731     78,730      3,293
Adjusted R2        0.228      0.235      0.504
-----------------------------------------------
```

This package also allows you to generate Latex output:

```
regtable(reg_w,reg_wf,reg_d; renderSettings = latexOutput())
# To create a Latex document with the table simply specify the name of the document:
# regtable(reg_w,reg_wf; renderSettings = latexOutput(table.tex))
```

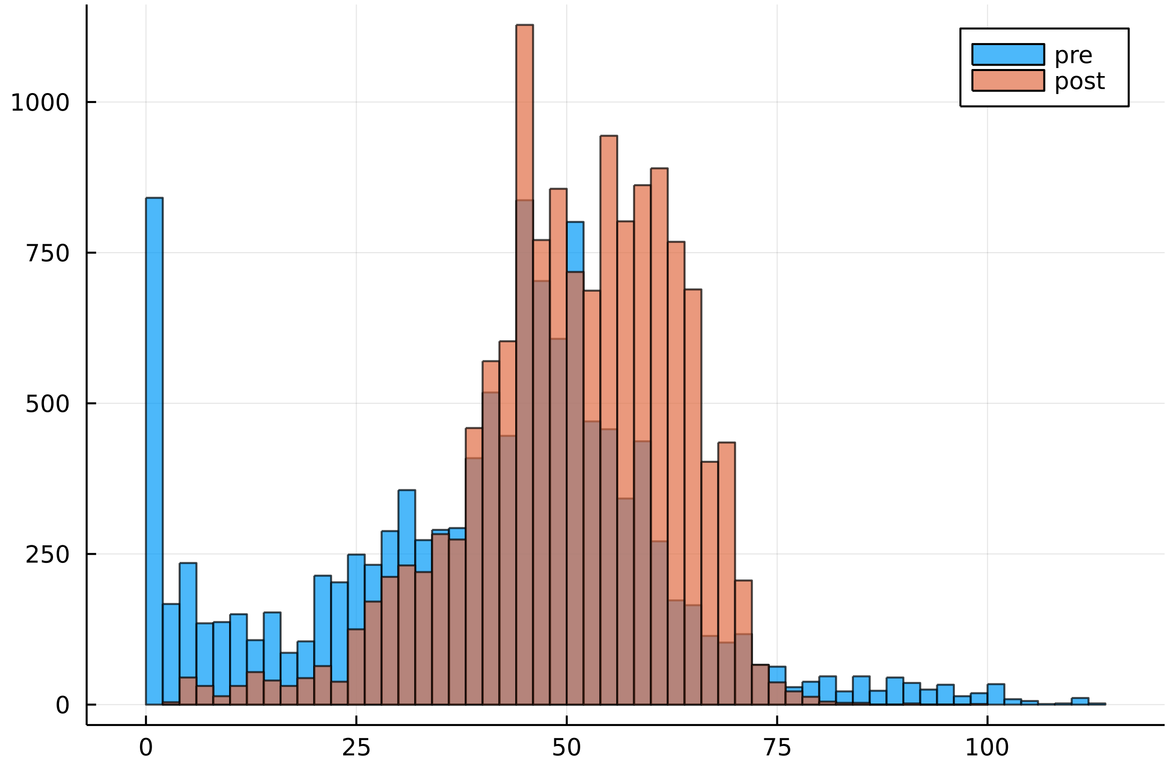
```
\begin{tabular}{lrrr}
\toprule
              & \multicolumn{3}{c}{system_costs} \\
\cmidrule(lr){2-4}
              &      (1) &      (2) &        (3) \\
\midrule
wind          & 0.187*** &          &            \\
              &  (0.004) &          &            \\
wind_forecast &          & 0.225*** &   0.256*** \\
              &          &  (0.004) &    (0.011) \\
\midrule
year          &      Yes &      Yes &        Yes \\
month         &      Yes &      Yes &        Yes \\
\midrule
Estimator     &      OLS &      OLS &        OLS \\
\midrule
$N$           &   78,731 &   78,730 &      3,293 \\
$R^2$         &    0.228 &    0.235 &      0.507 \\
\bottomrule
\end{tabular}
```

A policy change

The paper explores the **role of market design** in affecting the value of wind. The market moved away from subsidies that are paid based on production to subsidies that are based on installed capacity (subject to minimum performance requirments).

In the wholesale market, this implies that renewables no longer have an incentive to produce when prices are very low, e.g., as in California or Texas, in which prices are often zero or negative.

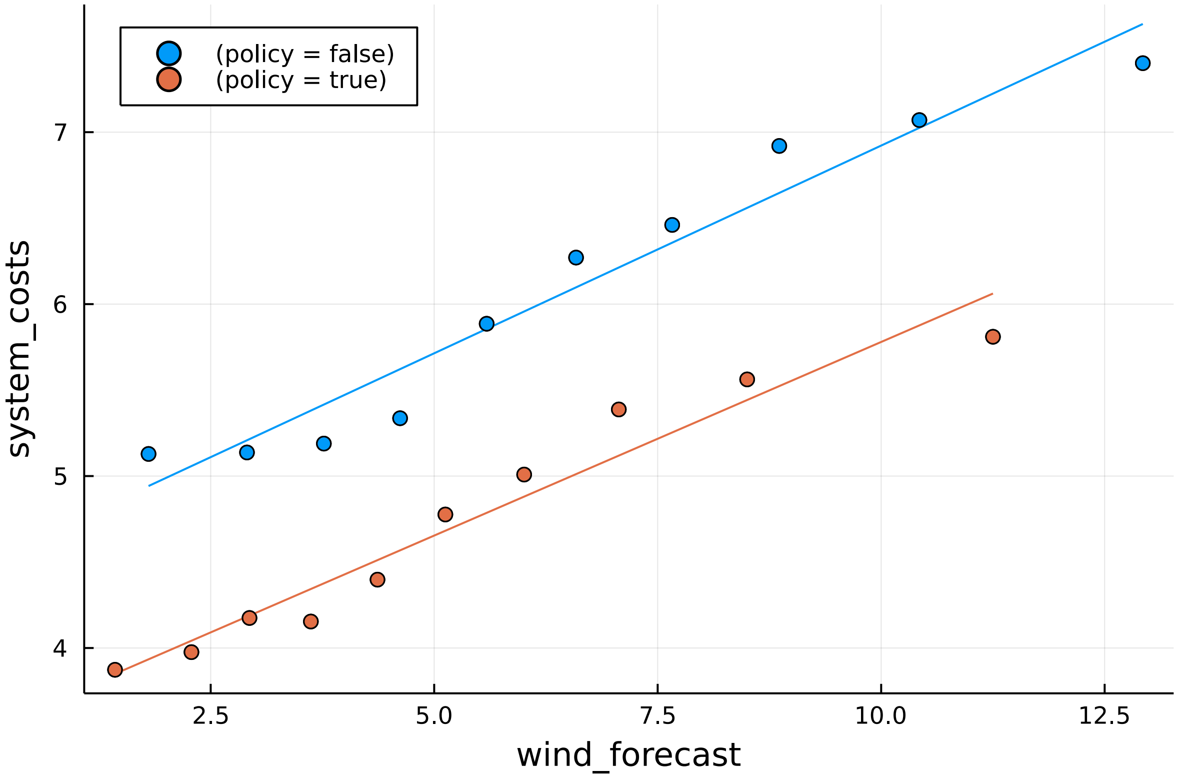
We will split the data in two to examine the change in the distribution of wholesale prices around the policy change.




```julia
begin
    df.policy = (((df.year .> 2014) .| ((df.year.==2014) .& (df.month .> 5))));
    df_policy = filter(row -> 2012 < row.year < 2016 ,df);

    histogram(df_policy.wholesale_price, group = df_policy.policy,
    alpha = 0.7,
    label = ["pre" "post"])
end
```

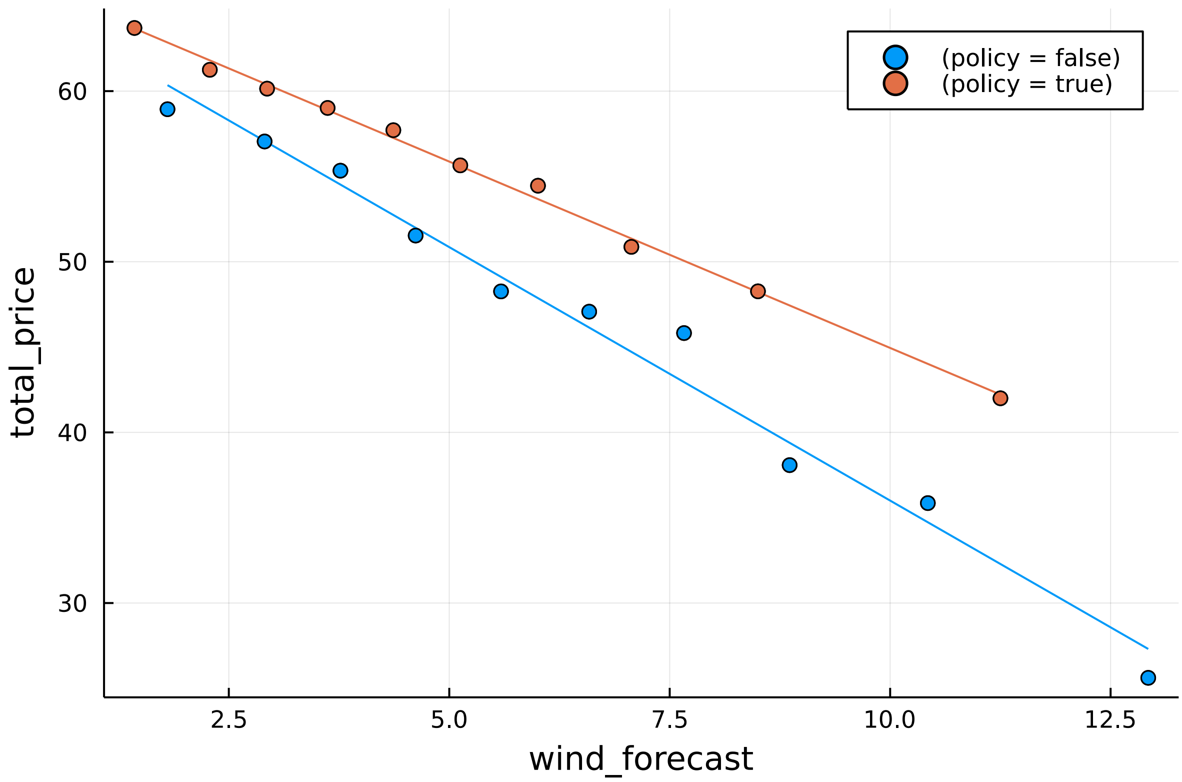
The policy change appeared to reduce system costs in the market. This could be due to the challenges of dispatching the market in the presence of zero prices, which lead to several strategic distortions.

We can plot system costs before and after the change.




```
begin
binscatter(groupby(df_policy, :policy), @formula(system_costs ~ wind_forecast), 10,
     seriestype = :linearfit,
        legend = :topleft)
end
```

Consumers were still worse off due to the increase in prices. Wind price reduction effect diminished.




```
begin
binscatter(groupby(df_policy, :policy), @formula(total_price ~ wind_forecast), 10,
    seriestype = :linearfit,
      legend = :topright)
end
```

Note: This is an event study, so there are other changes happening in the market. The idea here is to show major effects of the policy, but proper quantification requires more explicit control of confounders. To start with, although not exhaustive, we can estimate the effect of wind forecast after the policy change.

```
                                Fixed Effect Model
==================================================================================
Number of obs:                           78730  Degrees of freedom:             50
R2:                                      0.706  R2 Adjusted:                 0.706
F-Stat:                                19798.9  p-value:                     0.000
R2 within:                               0.502  Iterations:                      7
==================================================================================
wholesale_price          | Estimate  Std.Error  t value Pr(>|t|) Lower 95% Upper 95%
----------------------------------------------------------------------------------
wind_forecast            | -2.89202  0.0158909 -181.992    0.000  -2.92317  -2.86088
policy                   |   6.3441   0.235877  26.8958    0.000   5.88178   6.80642
demand_forecast          |  1.63373 0.00990177  164.994    0.000   1.61432   1.65314
wind_forecast & policy   | 0.817855  0.0212392  38.5069    0.000  0.776226  0.859484
==================================================================================
```

```
reg(df, @formula(wholesale_price ~ wind_forecast*policy + demand_forecast +
fe(hour) + fe(year) + fe(month)))
```

Follow-up exercises

1. What is the correlation of wind and demand? How could that affect the valuation of wind power?

2 (*). What is the environmental benefit of wind power in this market per unit of wind? Try to quantify that by regressing emissions on wind and converting it to a monetary amount using a valuation for emissions reductions. Estimate the total welfare effects of wind production. For that, you need to add to the environmental benefit the consumer and producer surplus. With respect to the producer surplus assume that the LCOE ranges between 50 to 90 €/MWh. How does your answer depend on the monetary value of reducing emissions?