Lianghui_report

Lianghui Li

January 21, 2018

Contact

*E-mail: ll291@duke.edu

*Phone: (510)600-9327

Report

Loading Data

```
power = read.csv("PowerCurve.csv")
site_df = read.csv("SiteData.csv")
wind = read.csv("WindData.csv")
energy = read.csv("SiteEnergy.csv")
```

Date Cleaning (~20 min)

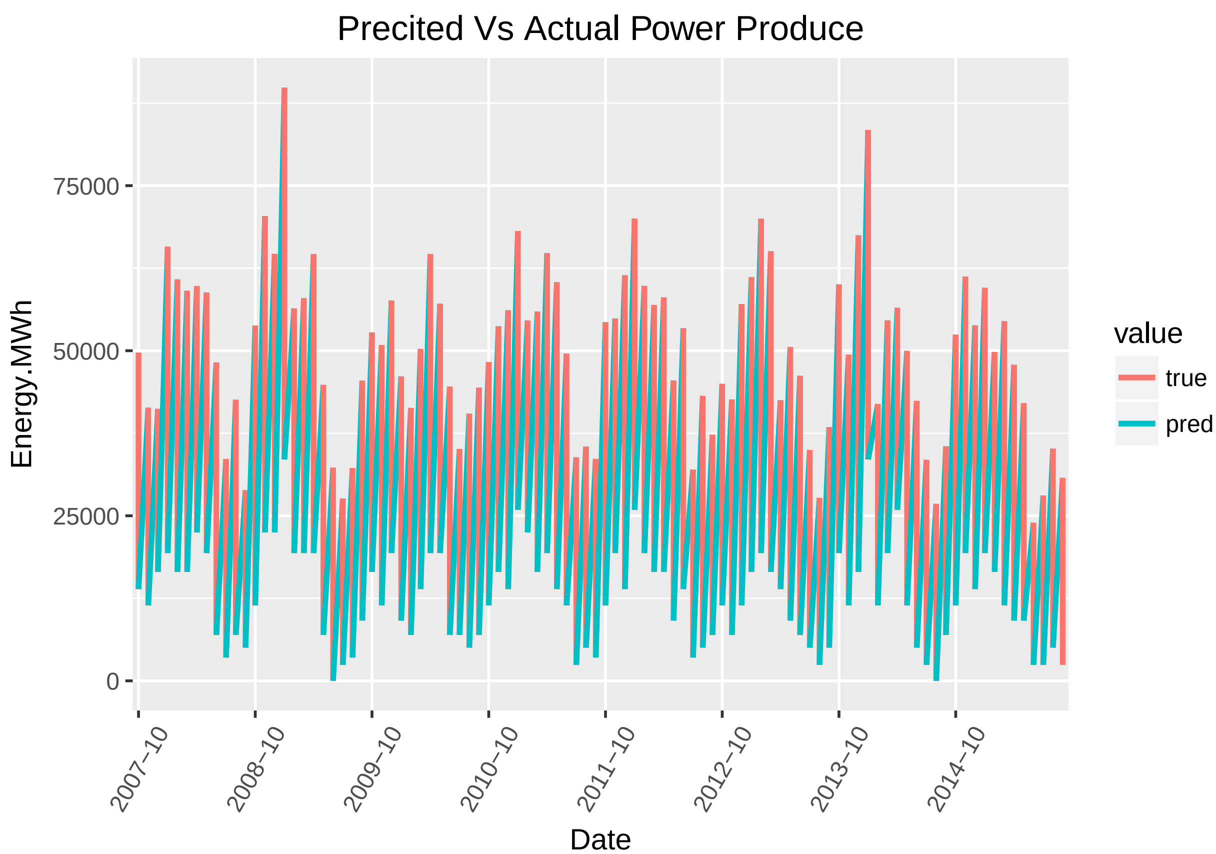
```
#filter the year within the range of avaible data(ie from 2017-10 to 2015-09)
year = as.numeric(substr(wind$Time,1,4))

year_filter = (year >= 2007 & year < 2016) & (!str_detect(wind$Time, "2015-1"))

#Aggregate hourly data to monthly and select useful columns for future use
#here I only choose 10m because of the turbine condition
wind_clean = wind[year_filter,] %>%
                      mutate(Month = str_extract(Time, "\\d{4}-\\d{2}")) %>%
                      group_by(Month) %>%
                      summarise(avg.windspeed = mean(NW_spd_10m),
                                avg.winddir = mean(NW_dir_10m)) %>%
                      merge(y = energy, by = "Month") %>%
                      select(-Capacity.MW) %>%
                      filter(!str_detect(.$Month,"2007-0"))

#create function that calculate predicted power generated from power curve
#since the table only provides certain amounts of corresonding wind speed(discrete)
#instead of a formula, my approach is to calculate the mean of power generate between the
#range of wind speed

assign_power = function(x) {
  y = c()
  wind_speed = power$WindSpeed
  energy_pred = power$Power.kW
```

```r
  for(i in seq_along(wind_speed)) {

    if (x > wind_speed[i] & x < wind_speed[i + 1]){

      y = mean(c(energy_pred[i], energy_pred[i+1]))}

    else if (x == wind_speed[i]) y = energy_pred[i]
    else if (x < 3.5 | x > 25)   y = 0
    }
  return(y)
}

#transform the number for all the turbines generating in a month (unit: MWh)
energy.pred = sapply(wind_clean$avg.windspeed, assign_power) * 134 * 30 * 24/1000

#create dataframe for next graph
df = data.frame(time = wind_clean$Month,
                true = wind_clean$Energy.MWh,
                pred = energy.pred)
df.long = melt(df, value.name = "value") %>%
                              mutate(time = as.character(time))
```

```
## Using time as id variables
```

Prediction Graph (~15 min)

```r
ggplot(df.long, aes(x = time, y = value, group = 1, col = variable)) +
  geom_line(size = 1) +
  scale_x_discrete(breaks = unique(df.long$time)[seq(1,nrow(df.long),12)]) +
  theme(axis.text.x  = element_text(angle=60, vjust=0.5)) +
  labs(title="Precited Vs Actual Power Produce",
       x ="Date", y = "Energy.MWh", col = "value") +
  theme(plot.title = element_text(hjust = 0.5))
```



Personally, since the true values are generally greater than my prediction, I think there are some computational errors when aggregating all the turbines or transforming the unit. In order to fully understand the relationship between wind speed and power generate, I decide to run a simple linear regression.

Linear Regression (~30min)

```
#first seperate dataset to train and test
set.seed(123)
train_index = sample(1:nrow(wind_clean), 0.5*nrow(wind_clean))
train = wind_clean[train_index,]
test = wind_clean[-train_index,]

#from power curve, the flexibility show cubic polynomial should be sufficient
mod1 = lm(Energy.MWh ~ avg.windspeed + I(avg.windspeed^2) + I(avg.windspeed^3) + avg.winddir,
   data = train)

pred.lm = predict(mod1, newdata = test)
pred.pow = energy.pred[-train_index]
```

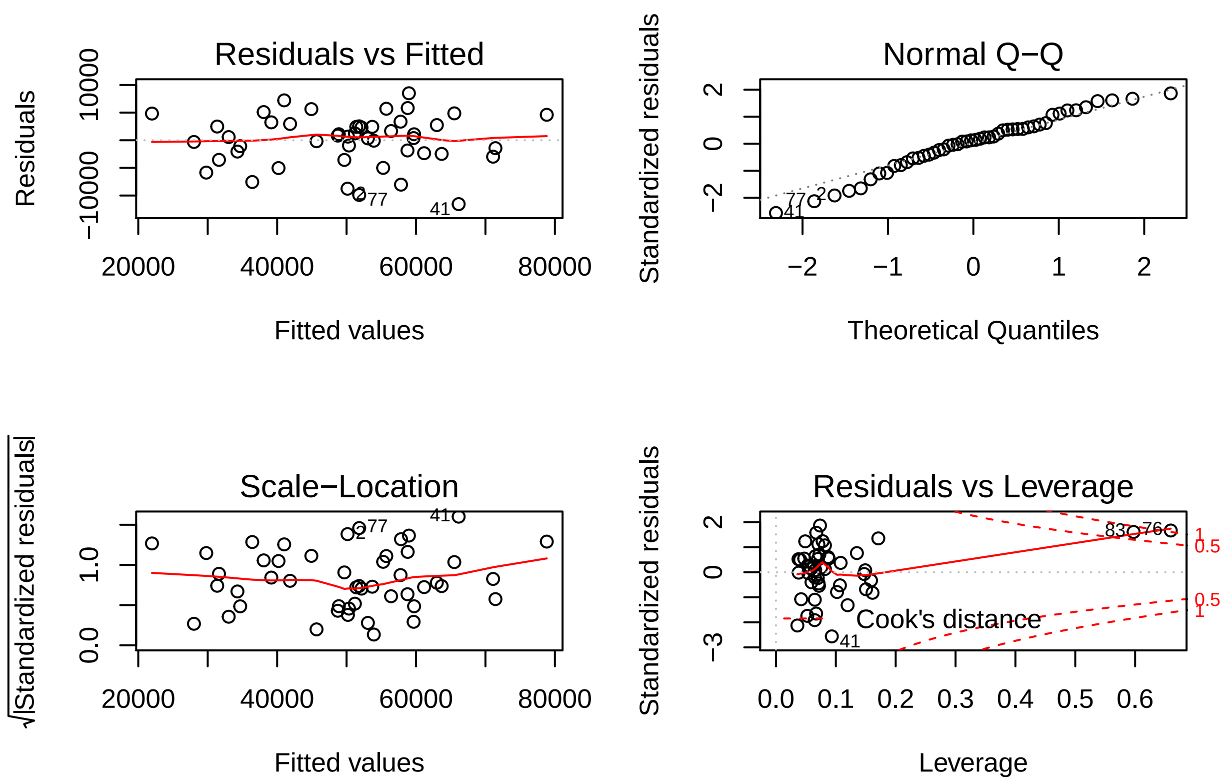
```
summary(mod1)
```

```
##
## Call:
## lm(formula = Energy.MWh ~ avg.windspeed + I(avg.windspeed^2) +
##     I(avg.windspeed^3) + avg.winddir, data = train)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11555.9  -2377.6    613.8   2793.7   8497.2
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -165820.29  132545.32  -1.251    0.218
## avg.windspeed         98182.19   82603.29   1.189    0.241
## I(avg.windspeed^2)   -16783.53   16989.55  -0.988    0.329
## I(avg.windspeed^3)     1106.40    1144.18   0.967    0.339
## avg.winddir              36.99      31.77   1.164    0.251
##
## Residual standard error: 4736 on 43 degrees of freedom
## Multiple R-squared:  0.8831, Adjusted R-squared:  0.8722
## F-statistic: 81.18 on 4 and 43 DF,  p-value: < 2.2e-16
```

We can see the R-squared(explained variable of response within the model), none of predictors are statistically significant.

```r
par(mfrow=c(2,2)) # Change the panel layout to 2 x 2
plot(mod1)
```




```r
par(mfrow=c(1,1)) # Change back to 1 x 1
```

The diagnostic plot for the linear model suggest some potential problems in this approach(ie, heteroscedasticity, residuals are not normally distrbuted and potential outliers and/or influential points), these prolems are potentially caused by underfitting(not enough features given in wind speed data)

```r
rmse = function(prediction) {
  rmse = sqrt(mean((test$Energy.MWh - prediction)^2))
  return(rmse)
}

df2 = data.frame(rmse(pred.lm),rmse(pred.pow))
colnames(df2) = c("linear_model", "power_curve")
kable(df2)
```

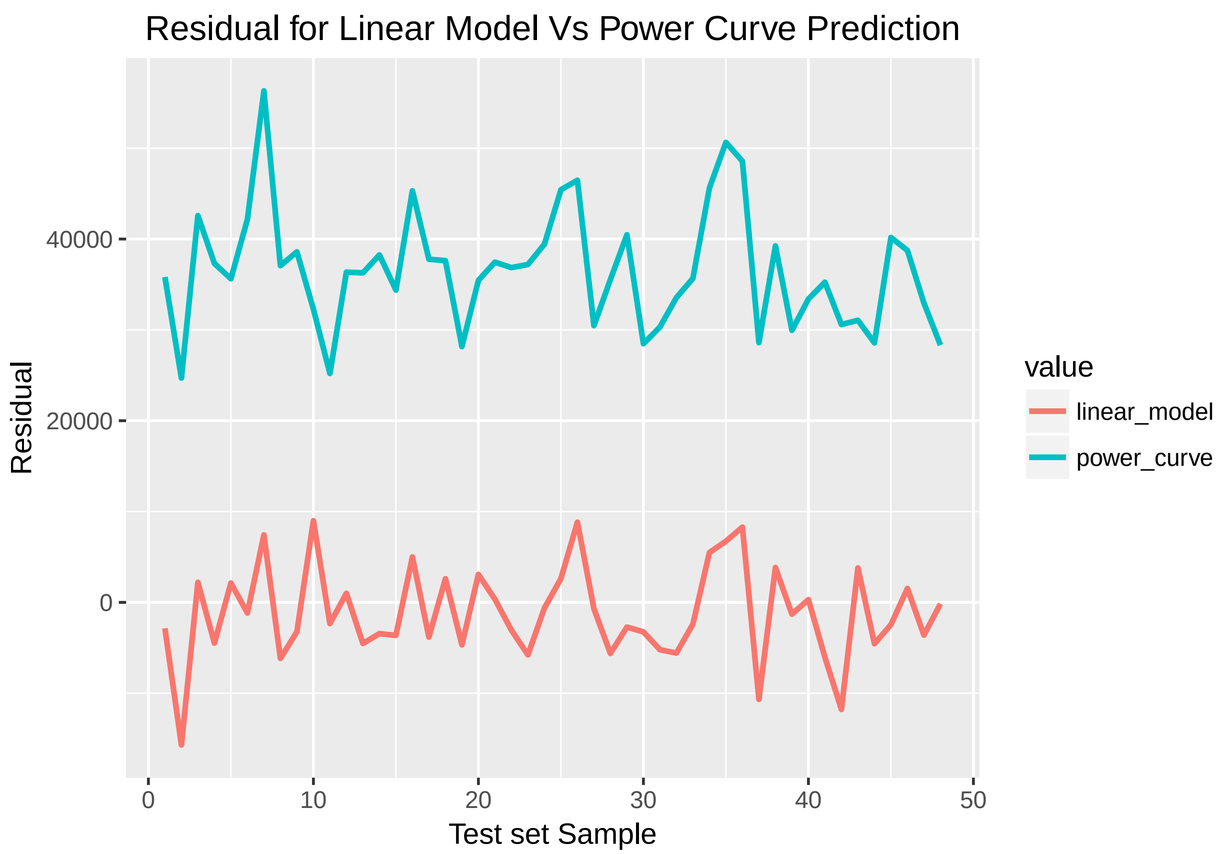
linear_model	power_curve
5301.324	37166.7

For the quantative results, I create a function calculating residual sum of squares:

$$RMSE = \sqrt{\frac{\sum_i^n (y_i - \hat{y}_i)^2}{n}}$$

As the table shown, linear model has better performance than the prediction from power curve in the test set.

```r
df3 = data.frame(index = 1:nrow(test),
                 linear_model = test$Energy.MWh - pred.lm,
                 power_curve = test$Energy.MWh - pred.pow)
df3.long = melt(df3, id = 'index', value.name = "value")

ggplot(df3.long, aes(x = index, y = value, col = variable)) +
  geom_line(size = 1) +
  labs(title="Residual for Linear Model Vs Power Curve Prediction",
       x ="Test set Sample", y = "Residual", col = "value") +
  theme(plot.title = element_text(hjust = 0.5))
```



Which can be shown in this plot as well.

To conclude, since there are issues with linear model in this case, and prediction from power curve are worse than linear model; I think the met data are not suitable to predict energy output at the site. (However, I believe I have made a major mistakes on the prediction from the power curve.)

Comment:

Overall I think it is a very interesting exercise, it gives me a chance to research in this industry and the topic; I was hoping to have more guidelines on determining how good the prediction should be for the model to be considered as suitable in this case (any industry standard?) and more background information will be helpful as well. I plan to use other machine learning model(like regression tree) to improve the result, but I am not sure if that is purpose of this assignment.