wind\_curve\_analysis\_month\_wise.R

sumedh

Mon Oct 08 20:59:24 2018

setwd("C:/Users/sumedh/Desktop/assignment/wind month wise curve")  
# Note I have rename the wtgdata to wt\_data  
wtdata=read.csv("wt\_data.csv")  
wtdata=na.omit(wtdata)  
library(WindCurves)# For drawing the Power curves

## Warning: package 'WindCurves' was built under R version 3.5.1

#libraries  
library(MASS)  
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.5.1

## -- Attaching packages ---------------------------------- tidyverse 1.2.1 --

## v ggplot2 2.2.1 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.5  
## v tidyr 0.8.1 v stringr 1.3.1  
## v readr 1.1.1 v forcats 0.3.0

## -- Conflicts ------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()  
## x dplyr::select() masks MASS::select()

library(knitr)

## Warning: package 'knitr' was built under R version 3.5.1

library(viridis)

## Loading required package: viridisLite

library(dplyr)  
library(scales)

## Warning: package 'scales' was built under R version 3.5.1

##   
## Attaching package: 'scales'

## The following object is masked from 'package:viridis':  
##   
## viridis\_pal

## The following object is masked from 'package:purrr':  
##   
## discard

## The following object is masked from 'package:readr':  
##   
## col\_factor

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':  
##   
## date

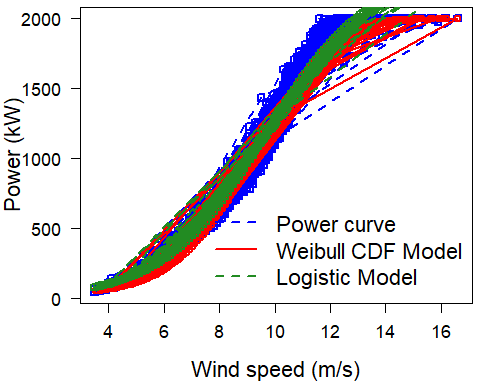
#### Converting into the data time format and storing it into the respective columns  
wtdata$ttimestamplocal=dmy\_hm(wtdata$ttimestamplocal)  
wtdata$month=month(wtdata$ttimestamplocal)  
wtdata$year=year(wtdata$ttimestamplocal)  
wtdata$day=day(wtdata$ttimestamplocal)  
wtdata$date=date(wtdata$ttimestamplocal)  
  
## filter the data month wise and wind turbine state='ok'  
dfjune = wtdata %>% filter(month==6,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
dfjuly = wtdata %>% filter(month==7,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
dfaug = wtdata %>% filter(month==8,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
dfsep = wtdata %>% filter(month==9,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
dfoct = wtdata %>% filter(month==10,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
dfnov = wtdata %>% filter(month==11,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
dfdec = wtdata %>% filter(month==12,wtg\_state=='ok') %>% select(windspeed,power,month,wtg\_state)  
  
## Analysis of Wind Curve month wise  
## June month  
  
s\_dfjune=dfjune$windspeed  
p\_dfjune=dfjune$power  
da\_dfjune=data.frame(s\_dfjune,p\_dfjune)  
x\_dfjune=fitcurve(da\_dfjune)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 4.755548   
## Scale (C) = 10.01027   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2342.764   
## phi 2 = 9.664736   
## phi 3 = 1.852823   
## ===================================

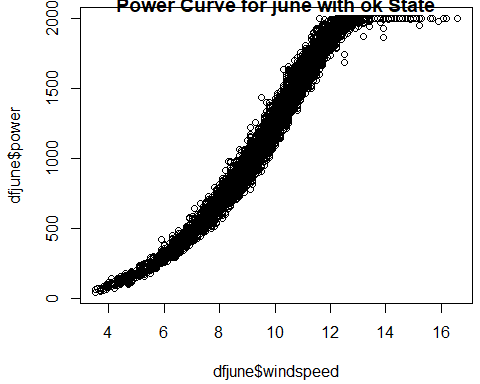
validate.curve(x\_dfjune)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 83.0961247 69.9377376  
## 2 MAE 69.6106916 54.1217032  
## 3 MAPE 11.2869148 5.5246234  
## 4 R2 0.9773145 0.9839302  
## 5 COR 0.9911493 0.9919530

plot(x\_dfjune)



plot(dfjune$windspeed,dfjune$power, main="Power Curve for june with ok State")



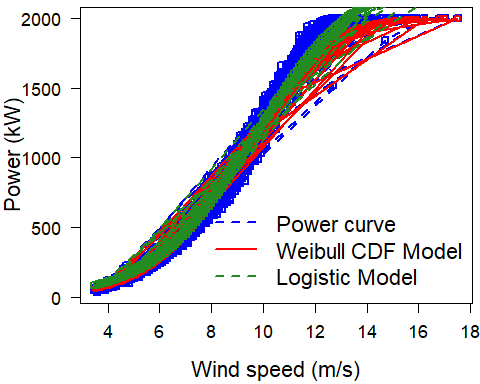
# For July Month  
s\_dfjuly=dfjuly$windspeed  
p\_dfjuly=dfjuly$power  
da\_dfjuly=data.frame(s\_dfjuly,p\_dfjuly)  
x\_dfjuly=fitcurve(da\_dfjuly)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 3.931009   
## Scale (C) = 10.04974   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2316.866   
## phi 2 = 9.674186   
## phi 3 = 1.893599   
## ===================================

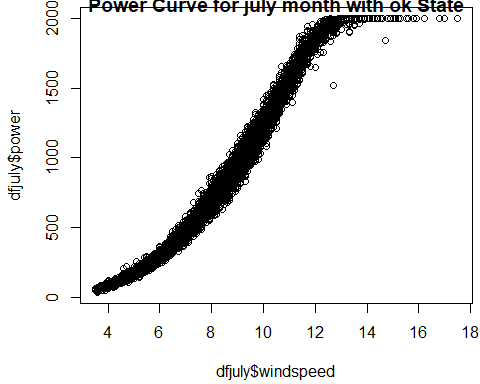
validate.curve(x\_dfjuly)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 63.9994688 56.4770530  
## 2 MAE 48.2740011 42.7356612  
## 3 MAPE 6.6949650 6.3885162  
## 4 R2 0.9859279 0.9890415  
## 5 COR 0.9933266 0.9945059

plot(x\_dfjuly)



plot(dfjuly$windspeed,dfjuly$power, main="Power Curve for july month with ok State")



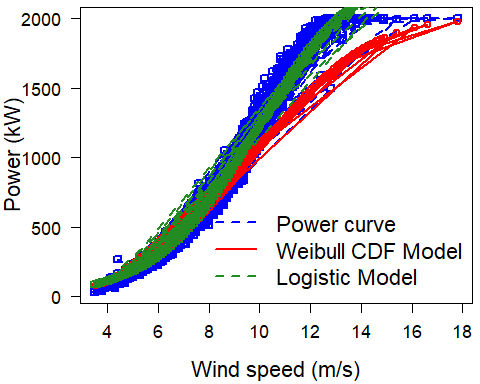
## For August Month  
s\_dfaug=dfaug$windspeed  
p\_dfaug=dfaug$power  
da\_dfaug=data.frame(s\_dfaug,p\_dfaug)  
x\_dfaug=fitcurve(da\_dfaug)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 3.175558   
## Scale (C) = 10.9262   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2383.641   
## phi 2 = 9.839721   
## phi 3 = 1.958011   
## ===================================

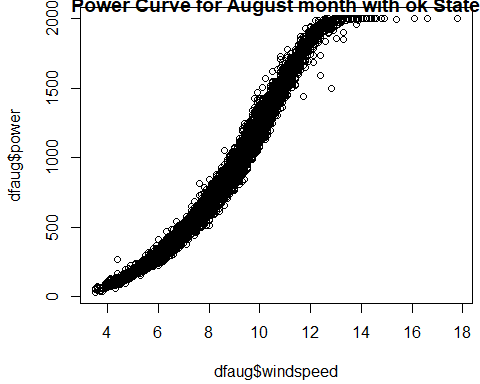
validate.curve(x\_dfaug)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 143.0745763 57.5528398  
## 2 MAE 98.9478318 43.6290774  
## 3 MAPE 11.0686433 6.1042369  
## 4 R2 0.9108819 0.9855797  
## 5 COR 0.9912189 0.9927641

plot(x\_dfaug)



plot(dfaug$windspeed,dfaug$power, main="Power Curve for August month with ok State")



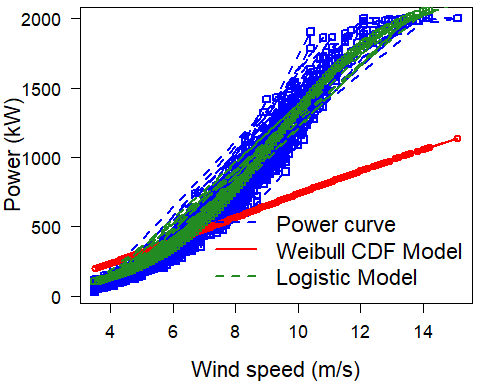
# For Sep month  
s\_dfsep=dfsep$windspeed  
p\_dfsep=dfsep$power  
da\_dfsep=data.frame(s\_dfsep,p\_dfsep)  
x\_dfsep=fitcurve(da\_dfsep)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 1.497477   
## Scale (C) = 17.13552   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2211.102   
## phi 2 = 9.188975   
## phi 3 = 1.903093   
## ===================================

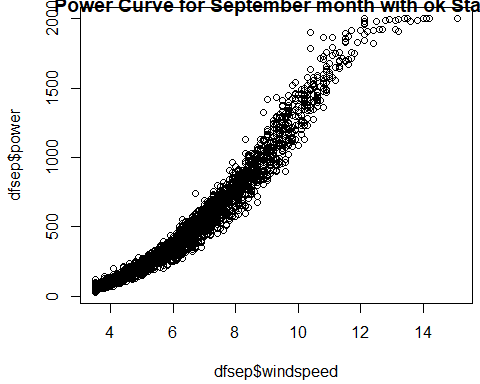
validate.curve(x\_dfsep)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 210.4678524 54.2006919  
## 2 MAE 136.5201589 37.0675967  
## 3 MAPE 31.5418093 10.0985082  
## 4 R2 0.6285014 0.9753626  
## 5 COR 0.9718781 0.9876593

plot(x\_dfsep)



plot(dfsep$windspeed,dfsep$power, main="Power Curve for September month with ok State")



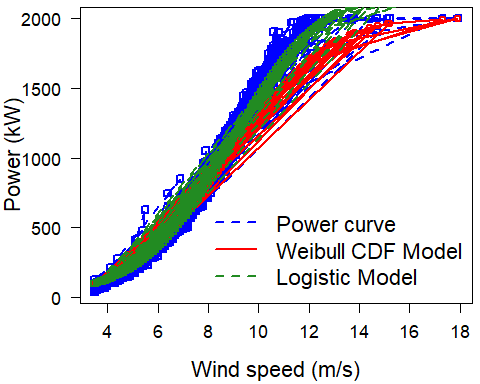
# For october Month  
s\_dfoct=dfoct$windspeed  
p\_dfoct=dfoct$power  
da\_dfoct=data.frame(s\_dfoct,p\_dfoct)  
x\_dfoct=fitcurve(da\_dfoct)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 3.288442   
## Scale (C) = 9.882469   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2231.938   
## phi 2 = 8.914355   
## phi 3 = 1.780752   
## ===================================

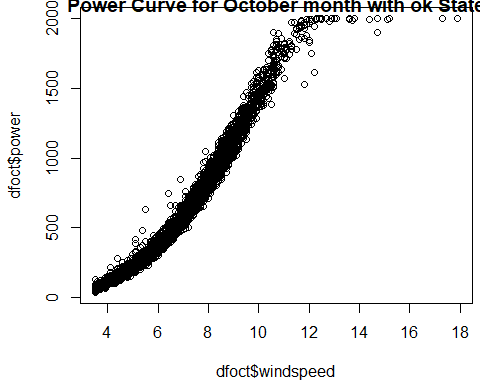
validate.curve(x\_dfoct)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 68.6309831 46.4110556  
## 2 MAE 44.3918030 33.6223987  
## 3 MAPE 8.1905198 7.0420264  
## 4 R2 0.9712989 0.9868750  
## 5 COR 0.9926321 0.9934245

plot(x\_dfoct)



plot(dfoct$windspeed,dfoct$power, main="Power Curve for October month with ok State")



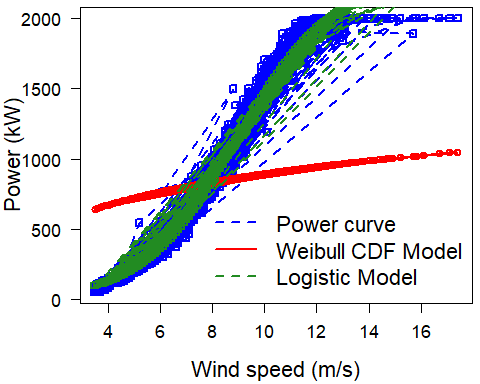
# For November month  
  
s\_dfnov=dfnov$windspeed  
p\_dfnov=dfnov$power  
da\_dfnov=data.frame(s\_dfnov,p\_dfnov)  
x\_dfnov=fitcurve(da\_dfnov)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 0.4250974   
## Scale (C) = 37.89628   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2278.768   
## phi 2 = 8.944602   
## phi 3 = 1.77267   
## ===================================

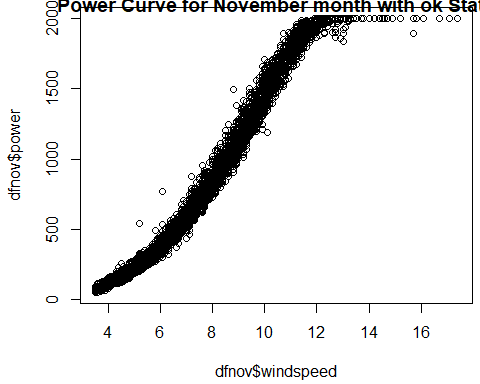
validate.curve(x\_dfnov)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 485.1739001 56.3180157  
## 2 MAE 407.1957264 41.2457303  
## 3 MAPE 49.3048517 6.1974175  
## 4 R2 0.2348225 0.9896899  
## 5 COR 0.9611049 0.9948319

plot(x\_dfnov)



plot(dfnov$windspeed,dfnov$power, main="Power Curve for November month with ok State")



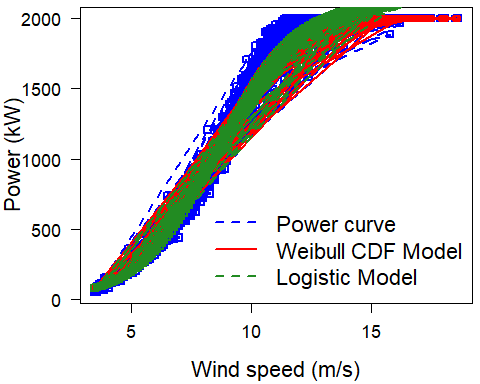
# For December month  
s\_dfdec=dfdec$windspeed  
p\_dfdec=dfdec$power  
da\_dfdec=data.frame(s\_dfdec,p\_dfdec)  
x\_dfdec=fitcurve(da\_dfdec)

## Weibull CDF model  
## -----------------  
## P = 1 - exp[-(S/C)^k]  
## where P -> Power and S -> Speed   
##   
## Shape (k) = 4.066103   
## Scale (C) = 9.188243   
## ===================================  
##   
## Logistic Function model  
## -----------------------  
## P = phi1/(1+exp((phi2-S)/phi3))  
## where P -> Power and S -> Speed   
##   
## phi 1 = 2125.975   
## phi 2 = 8.612468   
## phi 3 = 1.576264   
## ===================================

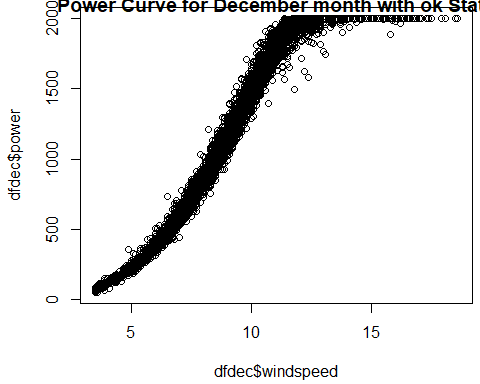
validate.curve(x\_dfdec)

## Metrics Weibull CDF Logistic Function  
## 1 RMSE 67.6203629 66.0065852  
## 2 MAE 51.6491235 52.4562348  
## 3 MAPE 4.8690243 5.2487126  
## 4 R2 0.9875842 0.9881698  
## 5 COR 0.9941451 0.9941209

plot(x\_dfdec)



plot(dfdec$windspeed,dfdec$power, main="Power Curve for December month with ok State")

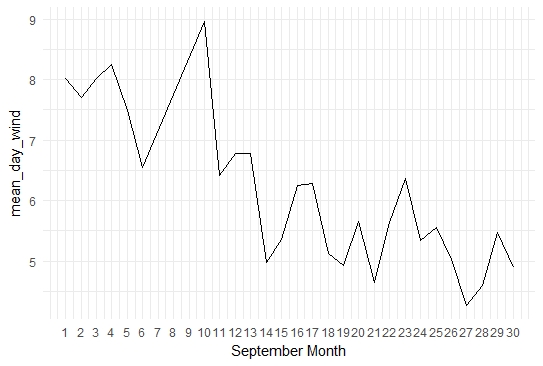


#### Further digging out the September Month as it is giving very poor MAPE Value for both wind Turbine

# For Wind Turbine 1

sept1=wtdata %>% filter(month==9,wtg\_state=='ok',unitlocation=='WTG01') %>% select(month,day,windspeed)%>% group\_by(day)%>% summarize(mean\_day\_wind=mean(windspeed))

ggplot(sept1,aes(day,mean\_day\_wind))+geom\_line()+theme\_minimal()+labs(x='day')+labs(x='September Month')+scale\_x\_continuous(breaks=seq(1,30,1))+scale\_y\_continuous(breaks=seq(1,12,1))+title(main = "In September Month Wind Turbine 1 with Ok state")



# Sep 14th,21st and 27th having the lowest windspeed in Wind Turbine 1 with Sep month and Ok State of Wind Turbine.

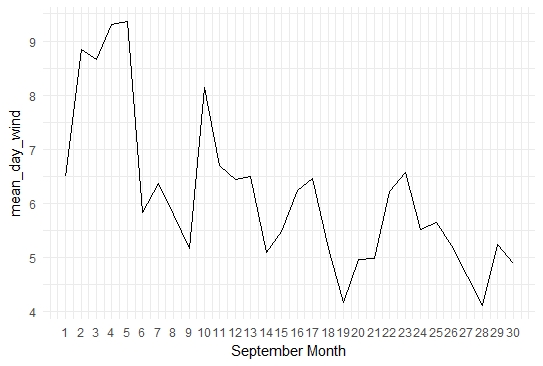
###################

# For wind turbine 2

sept2=wtdata %>% filter(month==9,wtg\_state=='ok',unitlocation=='WTG02') %>% select(month,day,windspeed)%>% group\_by(day)%>% summarize(mean\_day\_wind=mean(windspeed))

ggplot(sept2,aes(day,mean\_day\_wind))+

geom\_line()+theme\_minimal()+labs(x='day')+labs(x='September Month')+scale\_x\_continuous(breaks=seq(1,30,1))+scale\_y\_continuous(breaks=seq(1,12,1))+title(main = "In September Month Wind Turbine 2 with Ok state")



# In September month 9th,14th,19th,28th having the low value and data dips sudden on these dates with Wind turbine 2.