ST2005: Applied Probability II
Computing Laboratory 3

Task description

You have been approached to act as a statistical modelling consultant to an energy company who are reviewing operations at one of their power plants. They have provided you with data from the plant. The dataset records variables which the company's engineers believe are important factors in the operation of the plant. The company is interested in maximising net hourly electrical energy output (recorded as PE in the dataset). For each hour of energy output recorded, four other variables were recorded for that hour:

- Temperature (AT) in the range 1.81°C and 37.11°C
- Ambient Pressure (AP) in the range 992.89-1033.30 millibar
- Relative Humidity (RH) in the range 25.56% to 100.16%
- Exhaust Vacuum (V) in the range 25.36-81.56 cm Hg.

The company are particularly interested in how these variables affect the net hourly electrical energy output. The data contains these measurements over 9568 hours collected on randomly selected days over 6 years.

Reading in the dataset

Download the dataset *power_plant.csv* from blackboard. Save the dataset into your personal drive in the folder you have set up for this module. A `csv` file has ***c**omma **s**eparated **v**alues*, meaning the numerical values in a row of the dataset are separated by commas. Such files can be opened in a text editor or excel.

In order to read datasets in R, we must tell R exactly where the dataset lives (in the computer's directory). We can do this by setting the working directory (Session tab in R studio) to the dataset's location (your folder for this module).

The dataset can be read into an R data frame by using the *read.csv* function in R. Try *?read.csv* to bring up the help file for the function.

```
ppdat <- read.csv( file="power_plant.csv" )
```

Print out the first few lines of the dataset by using *head(ppdat)*. Find the names of the variables in the dataset by using *names(ppdat)*. Any variable in the data frame can be accessed using the *$* operator. For example, *ppdat$PE* gives the values of hourly energy output.

Plotting the data

The *pairs* function can be used to produce a matrix of scatter plots for the variables contained in a data frame. Get the help file using *?pairs*. Try to apply the function to your data frame. What patterns can you see? Change the plotting symbol, by passing the argument *pch=20* to the pairs function. Change the colour of the points by using *col="blue"*.

A single scatter plot examining the relationship between PE and any of the other variables can be produced by using the plot function. We provide vectors of x and y coordinates to the plot function. Try

```
plot( x=ppdat$AT, y=ppdat$PE )
```

Add x-axis and y-axis and title labels by using the *xlab*, *ylab* and *main* arguments to the plot function. Try changing the plotting symbol and the colour of the points. Try *?plot* to find out more.

Fitting a regression model

We can fit a (multiple) linear regression model using the *lm* function. The *lm* function takes a "formula" as an argument. If we wanted to fit a simple linear regression model predicting mean hourly energy output on ambient temperature, we could use

```
mod.fit <- lm( PE ~ AT, data=ppdat )
```

The formula is the first argument in the lm function with the "~". We could include additional predictor variables, say Ambient Pressure, in the model fit by appending the formula with "+ AP".

We can check the model output by using

```
summary( mod.fit )
```

Look at what the summary function outputs. What do the table of values mean. What are the p-values on the rightmost column of the table testing?

Fit a multiple regression model for hourly electrical energy output using all the possible predictor variables recorded in the dataset. Check which of the variables are significant predictors of the mean hourly energy output. Should any variables be excluded?