Computer Lab Week 9: solutions

STAT221

In this computer lab we will use a dataset on the chemical composition of wines. This dataset can be downloaded here

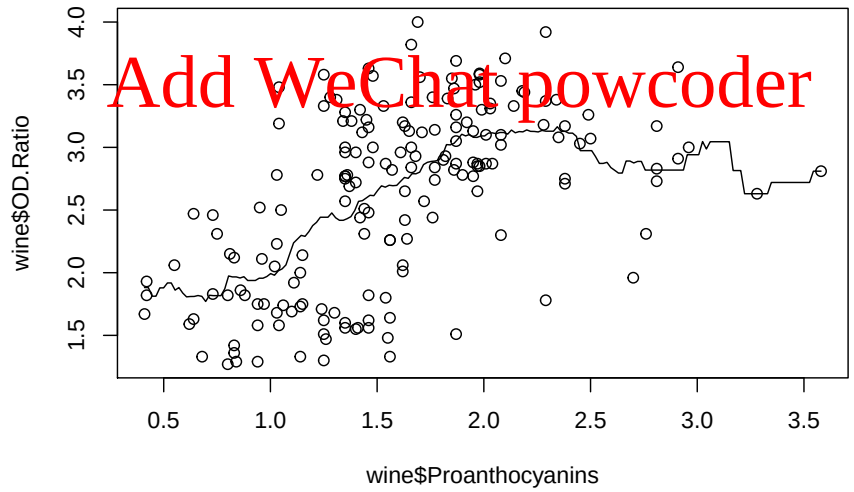
```
# Read Data into R Warning: whole address must be in one line
wine.fl = "http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data"
wine = read.csv(wine.fl, header = F)

# Names of the variables
wine.names = c("Alcohol", "Malic acid", "Ash", "Alcalinity of ash", "Magnesium",
               "Total phenols", "Flavanoids", "Nonflavanoid phenols", "Proanthocyanins",
               "Color intensity", "Hue", "OD.Ratio", "Proline")
colnames(wine)[2:14] = wine.names
colnames(wine)[1] = "Class"
```

Question 1.

Produce a scatterplot of `OD.Ratio` versus `Proanthocyanins` from the wine data set and superimpose onto it a line representing a box kernel smoother using the default bandwidth.

```
plot(wine$Proanthocyanins, wine$OD.Ratio)
lines(ksmooth(wine$Proanthocyanins, wine$OD.Ratio))
```

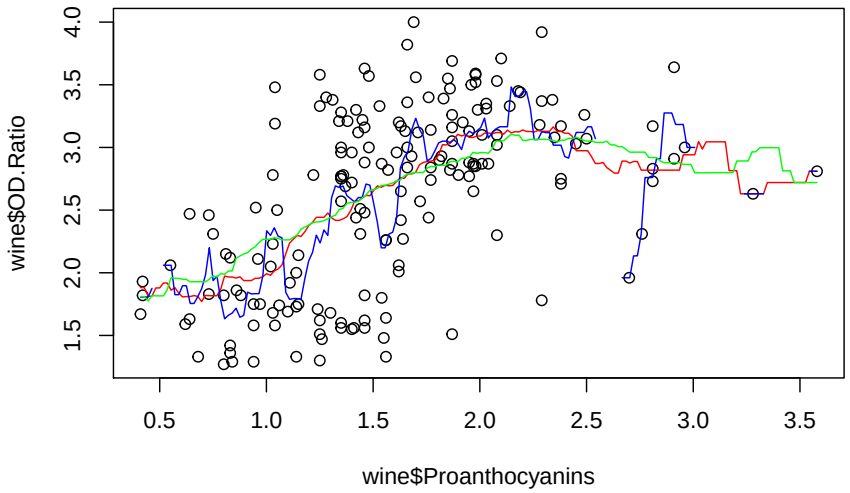
Comment: While the line looks 'choppy' at all points, it looks most choppy in areas where there is least data.

Question 2.

Look up the help page for `ksmooth`, and then play around with different values of the bandwidth to see if you can get a smoother or less choppy line, but still using the box kernel. Add these lines to the scatterp[lot using different colours for each line.]

```
plot(wine$Proanthocyanins, wine$OD.Ratio)
lines(ksmooth(wine$Proanthocyanins, wine$OD.Ratio), col = "red")
```

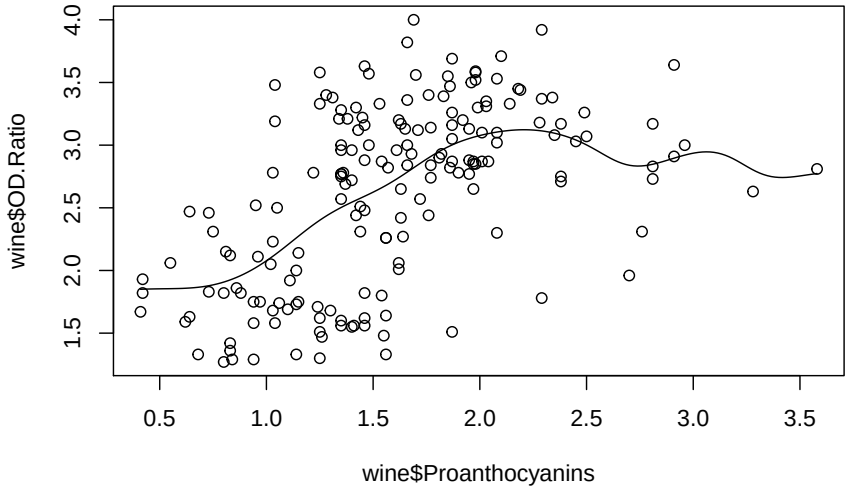
```
lines(ksmooth(wine$Proanthocyanins, wine$OD.Ratio, bandwidth = 0.1), col = "blue")
lines(ksmooth(wine$Proanthocyanins, wine$OD.Ratio, bandwidth = 1), col = "green")
```



Question 3.

Now switch to using the ''normal" kernel in `ksmooth`. Again produce a scatterplot of `OD.Ratio` versus `Proanthocyanins` and superimpose onto it a line representing a normal kernel smoother using the default bandwidth.


```
plot(wine$Proanthocyanins, wine$OD.Ratio)
lines(ksmooth(wine$Proanthocyanins, wine$OD.Ratio, "normal"))
```

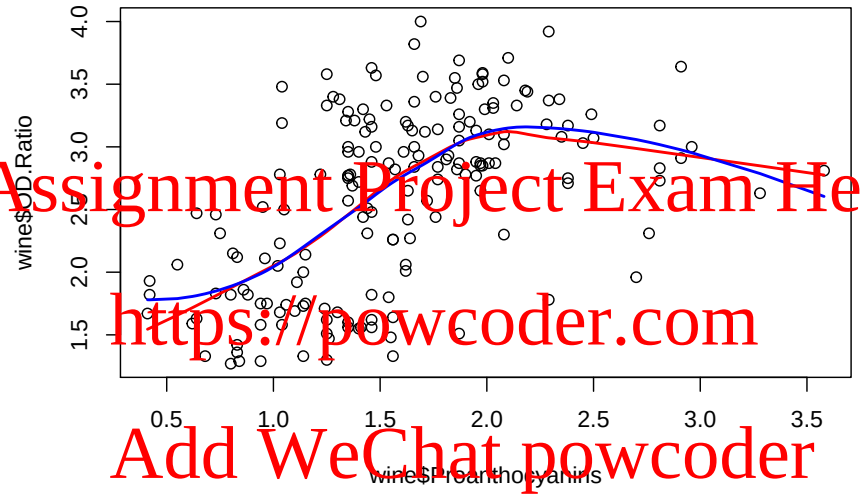
Comment: The line is now smooth, and we can see an almost linear relationship between the two variables up to a value of `Proanthocyanins` of around 2.0.

Question 4.

Now plot a default lowess smoother and a default loess smoother, again superimposed on a scatterplot of `OD.Ratio` versus `Proanthocyanins`.

```
# the lowess smoother
plot(wine$Proanthocyanins, wine$OD.Ratio)
lines(lowess(wine$Proanthocyanins, wine$OD.Ratio), col = "red", lwd = 2)

# the loess smoother
Proanthocyanins = sort(wine$Proanthocyanins)
OD.Ratio = wine$OD.Ratio[order(wine$Proanthocyanins)]
#plot(Proanthocyanins, OD.Ratio)
loess.fit = loess(OD.Ratio ~ Proanthocyanins)
lines(Proanthocyanins, predict(loess.fit), col = "blue", lwd = 2)
```

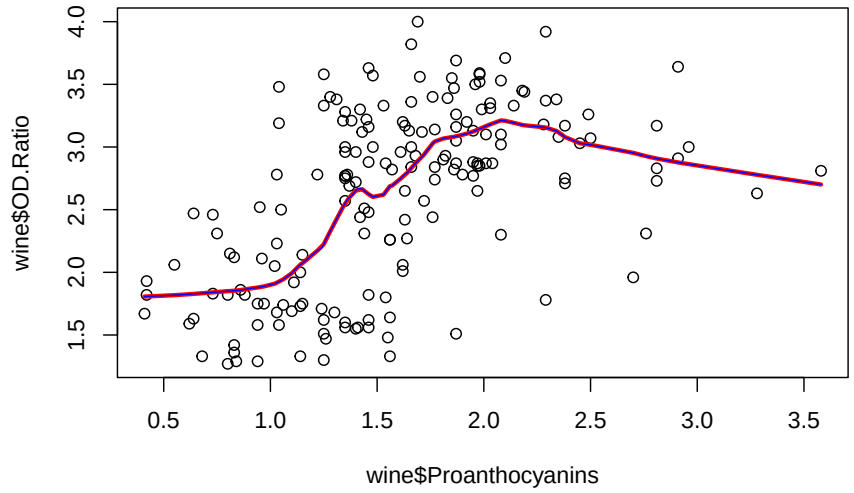


Why are they different? - think about what the defaults are for each of these smoothers.

Play around with the default settings and see if you can get the two lines to be identical: the idea here is just to play around with the different arguments and learn something about how they work, by having a goal of trying to get the lines to be the same. So don't worry too much if you can't get them to be identical, just give it a go.

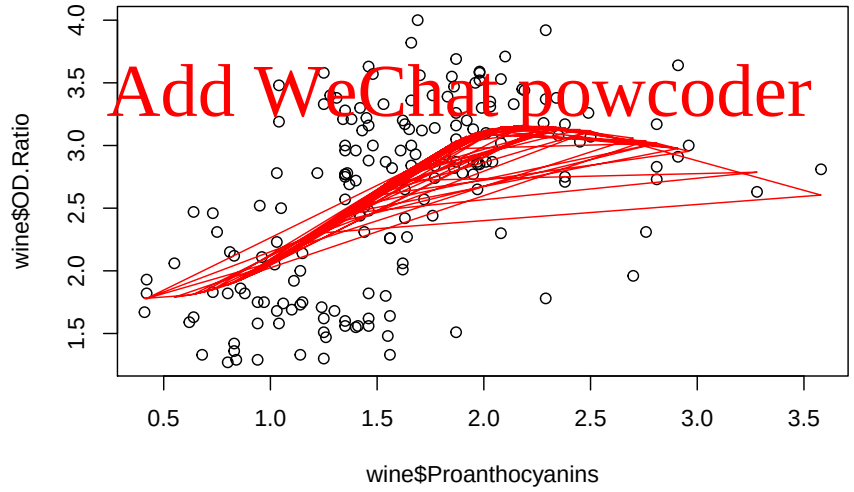
```
# the lowess smoother
plot(wine$Proanthocyanins, wine$OD.Ratio)
lines(lowess(wine$Proanthocyanins, wine$OD.Ratio, f = 0.3), col = "red", lwd = 3)

# the loess smoother
Proanthocyanins = sort(wine$Proanthocyanins)
OD.Ratio = wine$OD.Ratio[order(wine$Proanthocyanins)]
#plot(Proanthocyanins, OD.Ratio)
loess.fit = loess(OD.Ratio ~ Proanthocyanins,
                  span = 0.3, family = "symmetric", degree = 1, surface = "direct")
lines(Proanthocyanins, predict(loess.fit), col = "blue", lwd = 1)
```



If you get the output below for your loess smoother, then you have forgotten to sort your data into order so that R can draw a smooth curve using lines ...

```
plot(wine$Proanthocyanins, wine$OD.Ratio)
loess.fit = loess(wine$OD.Ratio ~ wine$Proanthocyanins)
lines(wine$Proanthocyanins, predict(loess.fit), col = "red")
```

Question 5.

Examine the effect of changing the bandwidth by producing 10 different plots arranged in a grid with two columns and five rows (using `mfrow`). For each of these 10 plots, produce a scatterplot of `OD.Ratio` versus `Proanthocyanins` and superimpose onto each one a single line representing a normal kernel smoother using the bandwidths 0.1, 0.2, ..., 1.0. Add a title to each plot that says what the bandwidth is of the smoother shown in that plot.

The plots are shown below to give you an idea of what you need to produce.

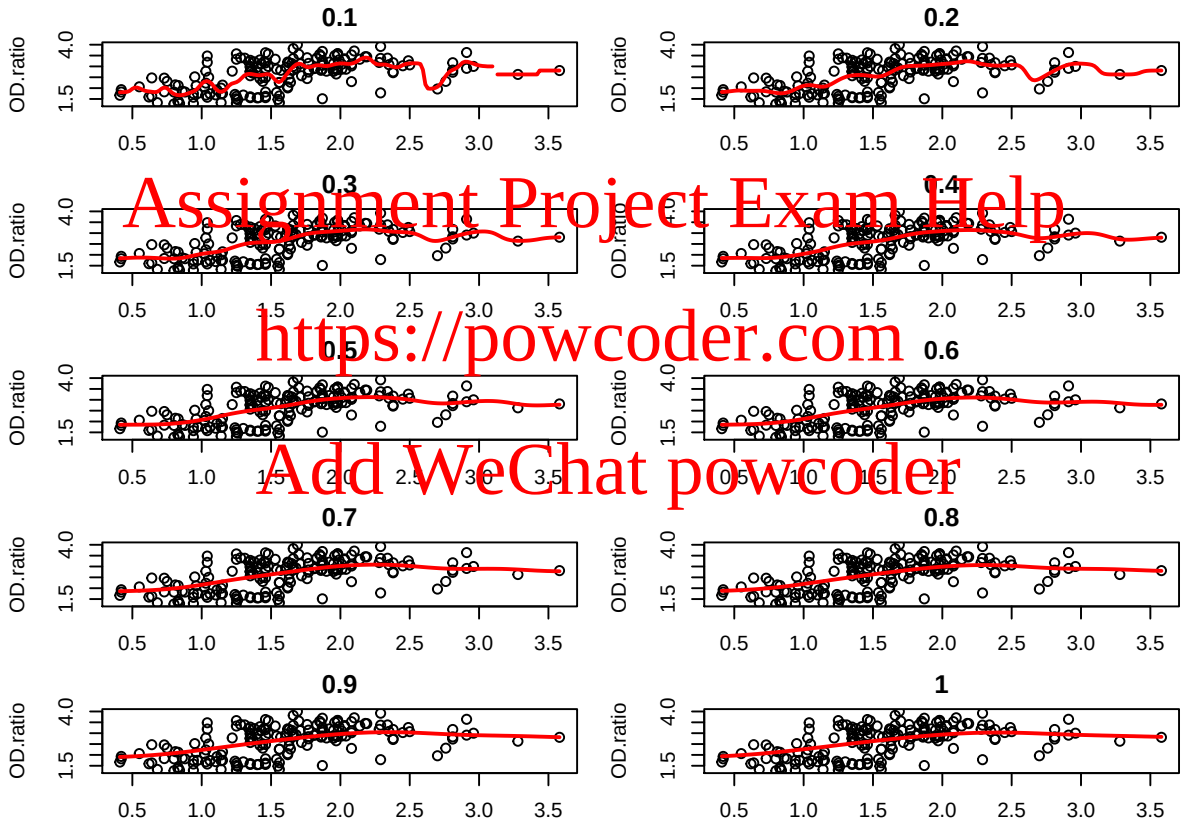
One approach is to loop around the code that produces a single plot, changing the bandwidth each time.

Another approach might be to write a function that produces a single plot and which has the bandwidth as its argument, and then run that function for each of the 10 bandwidths. The R function `lapply` might be useful for doing this.

Finally, it might be useful to play around with the margins around the outside of each of the 10 plots to make them narrower. Have a look at the help for `par` using `?par`, and then read about the graphical parameter called `mar`. You can adjust the default settings of `mar` to reduce the size of the margins. The default settings are `par(mfrow=c(1,1), mar=c(5,4,4,2))`.

```
kplotfn = function(bw){
  plot(wine$Proanthocyanins, wine$OD.Ratio, main=bw, ylab="OD.ratio")
  lines(ksmooth(wine$Proanthocyanins, wine$OD.Ratio, "normal", bandwidth=bw),
        col='red', lwd=2)
}

bws = seq(.1, 1, by = .1)
par(mfrow=c(5,2), mar=c(2,4,2,1)+.1)
lapply(bws, kplotfn)
```




```
par(mfrow=c(1,1))
```