Correlation in R

Use USairpollution data set.

1. Create scatterplots
	1. Basic r function
		1. **plot (X.variable, Y.variable, xlab=“x axis label”, ylab=“y axis label”)**
	2. with a regression fit line. You MUST run this with the plot function for it to work.
		1. **abline(lm(Y.variable ~ X.variable))**
	3. With enhanced features using a function from the car package
		1. **scatterplot(Y.variable ~ X.variable | grouping.variable, data=mydata,**

 **xlab=" x axis label ", ylab=" y axis label ", legend.title="group variable title", labels=row.names(mydata), smoother=F, boxplots="xy", jitter=list("xy"), xlim=c(40,80), grid=F)**

* + 1. legend.title to set the title of the grouping variable
		2. smoother indicates whether to include a smooth line (default it T=show line, use F to not show)
		3. reg.line indicates whether to include a regression line (default it T=show line, use F to not show)
		4. boxplots to add boxplots outside the matrix, for x variable, y variable, or both variables (xy)
		5. lwd=width of linear regression line (default is 1)
		6. jitter makes all data points visible
		7. xlim and ylim to change the limits of the axes
		8. legend.coords to specify where the legend for the grouping vaiable should be - options are “bottomright”, “bottom”, “bottomleft”, “left”, “topleft”, “top”, “topright”, “right”, “center”. Default is above the figure on the left
		9. smoother indicates whether to include a smooth line (default it T=show line, use F to not show)
		10. reg.line indicates whether to include a regression line (default it T=show line, use F to not show)
		11. by.groups=T to get separate regression lines per grouping variable
		12. lwd to set the width of linear regression line (default is 1)
		13. grid use F to not show the background grid
	1. Basic scatterplot matrix
		1. **scatterplotMatrix(~ variable1 + variable2 + variable3 + variable4, data=mydata, labels=row.names(mydata))**
	2. Basic catterplot matrix with grouping variable
		1. **scatterplotMatrix(~ variable1 + variable2 + variable3 + variable4 | grouping.variable, data=mydata, labels=row.names(mydata))**
	3. Scatterplot matrix with options
		1. **scatterplotMatrix(~ variable1 + variable2 + variable3 + variable4 | grouping.variable, data=mydata, labels=row.names(mydata), var.labels=c("variable1.name", " variable2.name ", " variable3.name ", " variable4.name "), diagonal=c("histogram"), nclass=6, smoother=F, by.groups=T, lwd=2)**
		2. See 1c for options also available for a scatterplot
		3. diagonal: what to plot on the diagonal – options are density, boxplot, histogram, qqplot, none (density is the default)
		4. nclass to set the number of levels for the histograms
1. Calculate the covariance and correlation for two variables
	1. **cov(temp, precip)**
	2. **cor.test(temp, precip)**
		1. Also returns df, p-value, and confidence interval. The p-value also applies to the covariance.
2. Calculate covariance and correlations for all variables in a dataframe or matrix
	1. **cov(USairpollution, use="pairwise.complete.obs")**
	2. **cor(USairpollution, use="pairwise.complete.obs", method="pearson")**
	3. use: how to handle missing data, other options are all.obs (for no missing data) and complete.obs (listwise deletion)
	4. method: type of correlation, other options are spearman and kendall
3. Compute correlations between certain variables (not a whole matrix)
	1. **x <- USairpollution[1:2]**: these variables will be the rows; the numbers indicate variables 1 and 2 from the data set
	2. **y <- USairpollution[3:7]**: these variables will be the columns; the numbers indicate variables 3 through 7
	3. **cov(x,y)**
	4. **cor(x,y)**
4. Calculate covariance and correlation for different levels of a categorical variable using a function in the plyr package
	1. **ddply(mydata, .(categorical.variable), summarise, "covariance" = cor(variable1, variable2))**
	2. ddply = input is a dataframe and output is a dataframe
	3. replace “covariance” with “corr” to output correlations
5. Compare correlations with a function from the psych package
	1. independent correlations
		1. **paired.r(correlation1, correlation2, NULL, sample.size, n2=samlple.size2)**
		2. NULL is needed because there is not a third correlation
		3. n2= is optional and only needed when the groups are not the same size
		4. outputs the z and p values
	2. Compare dependent correlations
		1. **paired.r(r31, r32, r12, N**
		2. outputs the t and p values
6. Converting correlations
	1. From r to Fisher’s z: **fisherz(correlation value)**
	2. From Fisher’s z to r: **fisherz2r(z value)**
	3. From r to effect size d: **r2d(correlation value)**
7. Online calculator for p-values: <http://www.socscistatistics.com/pvalues/pearsondistribution.aspx>