Regression

Eric Feigelson (Penn State) edf@astro.psu.edu

2nd East Asian Workshops in Astrostatistics Summer 2018

Adapted from R scripts in Appendix B, *Modern Statistical Methods for Astronomy With R Applications*, Eric D. Feigelson & G. Jogesh Babu 2012 http://astrostatistics.psu.edu/MSMA (http://astrostatistics.psu.edu/MSMA)

We first exercise one of the most widely used functions in R, lm = linear modeling, and related procedures. Our dataset is a collection of photometry of spectroscopically confirmed quasars from the Sloan Digital Sky Sruvey. We examine a relationship between the magnitdues in two bands; this is scientifically rather useless, but gives opportunity to test methodology for simple linear regression with difficulties common in astronomical regressions: non-Gaussian scatter, heteroscedastic measurement errors, and outliers.

In [ ]:
```
# I. Construct large and small samples of 77,429 SDSS quasars

setwd('/Users/ericfeigelson/Desktop/Rdir/')
qso_orig <- read.table(file='SDSS_QSO.dat', head=T)
names(qso_orig)
dim(qso_orig)
summary(qso_orig)
qso <- qso_orig[-which(qso_orig[,3] == 0 | qso_orig[,9] == 0),]# remov
e some bad photometry
qso[(qso[,4]<0.02),4 ] <- 0.02        # set threshold on magnitude err
ors
dim(qso) ; summary(qso)
attach(qso)
```

When applied to a data.frame, the R function `attach` allows the user to access a column by its names (e.g. r_mag) without remembering their column number (e.g. qso[,3]).

In [ ]:
```
# Plot dataset of SDSS quasar i vs. u magnitudes showing
# heteroscedastic measurement errors, with contours for dense regions

plot(i_mag, u_mag, pch=20, cex=0.1, col='#00000040', xlim=c(16,21),
   ylim=c(16.5,23.5), xlab="SDSS i (mag)", ylab="SDSS u (mag)")
for(i in 50:150) {
   lines(c(i_mag[i],i_mag[i]),c((u_mag[i]+sig_u_mag[i]),
      (u_mag[i]-sig_u_mag[i])), col='purple2')
   lines(c((i_mag[i]+sig_i_mag[i]),(i_mag[i]-sig_i_mag[i])),
      c(u_mag[i],u_mag[i]), col='purple2')   }

library(KernSmooth)
smqso <- bkde2D(cbind(i_mag, u_mag), bandwidth=c(0.05, 0.05), gridsize
=c(400,400))
contour(smqso$x1, smqso$x2, smqso$fhat, add=T, col='gold', nlevels=9)
```

Here we see a disturbed dataset. There is wide asymmetrical scatter towards main magnitudes in the SDSS u (ultraviolet) band. Much of the scatter is attributable to measurement errors, but not all of it.

In [ ]:
```
# II. Ordinary least squares fit

fit_ols <- lm(u_mag~i_mag)
summary(fit_ols)
confint(fit_ols, level=0.997)  # 3 sigma equilvalent for Gaussian dist
ribution
plot(i_mag, u_mag, pch=20, cex=0.1, col='#00000040', xlim=c(16,21),
   ylim=c(16.5,23.5), xlab="SDSS i (mag)", ylab="SDSS u (mag)")
abline(fit_ols$coef, lty=1, lwd=2)  # solid black line
```

Note that, since the scatter is non-Gaussian, the ordinary least squares fit is not a maximum likelihood estimator. The parameter uncertainties also may not be reliable. Visually, this is a terrible fit, missing most of the data points. We now try to improve it in three ways: weighting by measurement errors; applying `robust` downweighting of outliers; and applying both corrections.

In [ ]:
```
# III. Weighted least squares fit

fit_wt <- lm(u_mag~i_mag, x=T, weights=1/(sig_u_mag*sig_u_mag))
summary(fit_wt)

plot(i_mag, u_mag, pch=20, cex=0.1, col='#00000040', xlim=c(16,21),
   ylim=c(16.5,23.5), xlab="SDSS i (mag)", ylab="SDSS u (mag)")
abline(fit_wt$coef,lty=2,lwd=2, col='darkgreen')    # dashed dark-gree
n line

# IV. Robust M-estimator

library(MASS)
fit_M <- rlm(u_mag~i_mag, method='M')   # robust fit with Huber's psi
functon
summary(fit_M)
aM <- fit_M$coef[[1]] ; bM <- fit_M$coef[[2]]
lines(c(16,21), c(aM+bM*16, aM+bM*21), lty=3, lwd=3, col='royalblue3')
# dotted royal blue line

fit_Mwt <- rlm(u_mag~i_mag, method='M', weights=1/(sig_u_mag*sig_u_mag
),
   wt.method='inv.var')   # robust fit with measurement error weightin
g
summary(fit_Mwt)
aMwt <- fit_Mwt$coef[[1]] ; bMwt <- fit_Mwt$coef[[2]]
lines(c(16,21), c(aMwt+bMwt*16, aMwt+bMwt*21), lty=3, lwd=3, col='dark
blue')
text(19.5, 17, 'u = 0.13 + 1.02*i', cex=1.3, col='darkblue')
```

Here we see that most of the problems can be removed with measurement error weighting. Astronomers often do not carefully examine the accuracy and validity of their regression fits. Diagnostic graphics are very useful for this. Here are the plots produced automatically by R's `lm` function. For interpretation and details, see the text `A Modern Approach to Regression with R` (S. Sheather, 2009).

In [ ]:
```
# Diagnostic plots involving regression residuals help identify outlie
rs
par(mfrow=c(2,2))
plot(fit_wt, which=c(2:5), caption='', sub.caption='' ,pch=20, cex=0.3
,
   cex.lab=1.3, cex.axis=1.3)
```

Another approach to non-Gaussianity and outliers is to apply `robust` regression techniques. These are many variants; here R's `rlm` (robust linear modeling) function, downweighting outliers using Huber's psi function, with and without measurement error weighting. Unfortunately, this code does not have a built-in line plotting option, so we draw the lines manually from information in the `rlm` output. See various approaches in R at the CRAN Task View on Robust Statistics.

In [ ]:
```
# IV. Robust M-estimator

library(MASS)
fit_M <- rlm(u_mag~i_mag, method='M')   # robust fit with Huber's psi
functon
summary(fit_M)
aM <- fit_M$coef[[1]] ; bM <- fit_M$coef[[2]]

plot(i_mag, u_mag, pch=20, cex=0.1, col='#00000040', xlim=c(16,21),
   ylim=c(16.5,23.5), xlab="SDSS i (mag)", ylab="SDSS u (mag)")
lines(c(16,21), c(aM+bM*16, aM+bM*21), lty=3, lwd=3, col='royalblue3')
# dotted royal blue line

fit_Mwt <- rlm(u_mag~i_mag, method='M', weights=1/(sig_u_mag*sig_u_mag
),
   wt.method='inv.var')   # robust fit with measurement error weightin
g
summary(fit_Mwt)
aMwt <- fit_Mwt$coef[[1]] ; bMwt <- fit_Mwt$coef[[2]]

lines(c(16,21), c(aMwt+bMwt*16, aMwt+bMwt*21), lty=3, lwd=3, col='dark
blue')
text(19.5, 17, 'u = 0.13 + 1.02*i', cex=1.3, col='darkblue')
```

We now turn to nonlinear regression. Astronomers often fit data with nonlinear functions derived from astrophysical theory that we believe apply to the observed situation. These range from elliptical orbits for exoplanets, to the consensus Lambda-CDM model in cosmology. But we also often fit data with heuristic nonlinear functions; e.g. stellar Initial Mass Function, Schechter (gamma) galaxy luminosity function, Navarro-Frenk-White Dark Matter profile, etc.

Here we fit radial profiles from nearby Virgo Cluster elliptical galaxies to a heuristic nonlinear function proposed by Jose Luis Sersic in 1968. The data are obtained from Kormendy et al. (2009). We fit using R's `nls` (nonlinear least squares) function.

Exercise: Exercise options for nls using `nlscontrol` and `nlsModel`. Try fitting with CRAN packages `nmle` (nonlinear maximum likelihood estimation), `gnls` (generalized nonlienar least squares), and `gnm` (generalized nonlinear models).

V. Fit Sersic function to NGC 4472 elliptical galaxy surface brightness profile

In [ ]:
```
NGC4472 <- read.table("NGC4472_profile.dat", header=T)
attach(NGC4472)
NGC4472.fit <-  nls(surf_mag ~ -2.5*log10(I.e * 10^(-(0.868*n-0.142)*
   ((radius/r.e)^{1/n}-1))) + 26, data=NGC4472, start=list(I.e=20.,
   r.e=120.,n=4.), model=T, trace=T)
summary(NGC4472.fit)
logLik(NGC4472.fit)
```

Plot the result, along with similar fits to two other Virgo elliptical galaxies.

In [ ]:
```
# Plot NGC 4472 data and best-fit model

par(mai=c(1,1,0.8,0.44))   # improve left-hand margin
plot(NGC4472.fit$model$radius, NGC4472.fit$model$surf_mag, pch=20,
   xlab="r  (arcsec)", ylab=expression(mu ~~ (mag/sq.arcsec)), ylim=c(
16,28),
   cex.lab=1.5, cex.axis=1.5)
lines(NGC4472.fit$model$radius, fitted(NGC4472.fit))

# Fit and plot radial profiles of NGC 4406 and NGC 4451

NGC4406 <- read.table("NGC4406_profile.dat", header=T)
attach(NGC4406)
NGC4406.fit <- nls(surf_mag ~ -2.5*log10(I.e * 10^(-(0.868*n-0.142)*
   ((radius/r.e)^{1/n}-1))) + 32, data=NGC4406, start=list(I.e=20.,
   r.e=120.,n=4.), model=T, trace=T)
summary(NGC4406.fit)
points(NGC4406.fit$model$radius, NGC4406.fit$model$surf_mag, pch=3)
lines(NGC4406.fit$model$radius, fitted(NGC4406.fit))

NGC4551 <- read.table("NGC4551_profile.dat", header=T)
attach(NGC4551)
NGC4551.fit <-  nls(surf_mag ~ -2.5*log10(I.e * 10^(-(0.868*n-0.142)*
   ((radius/r.e)^{1/n}-1))) + 26, data=NGC4551, start=list(I.e=20.,r.e
=15.,n=4.),
   model=T, trace=T)
summary(NGC4551.fit)
points(NGC4551.fit$model$radius, NGC4551.fit$model$surf_mag, pch=5)
lines(NGC4551.fit$model$radius, fitted(NGC4551.fit))
legend(500, 20, c("NGC 4472","NGC 4406", "NGC 4551"), pch=c(20,3,5))
```

Print various scalar quantities from the `nls` fit, and plot the residuals between the data and model. A nonparametric smoother is added to assist seeing the amazing structure in the residuals: periodic shells of stars in excess of the monotonic Sersic model. This is a well-known effect due to past galaxy mergers that form large elliptical galaxies, and a similar residual plot appears in Kormendy's paper.

In [ ]:
```
# Details information about the nls fit

formula(NGC4472.fit)    # formula used
coef(NGC4472.fit)       # best-fit parameters
confint(NGC4472.fit)    # 95% confidence intervals
profile(NGC4472.fit)    # profiles (cuts) around the best fit
vcov(NGC4472.fit)       # best-fit parameter covariance matrix
logLik(NGC4472.fit)    # log-likelihood of best fit
fitted(NGC4472.fit)     # fitted values
residuals(NGC4472.fit)  # residuals from the fitted values

# Residual plot

plot(NGC4472.fit$model$radius,residuals(NGC4472.fit), xlab="r (arcsec)
",
   ylab="Residuals", pch=20, cex.lab=1.5, cex.axis=1.5)
lines(supsmu(NGC4472.fit$model$radius, residuals(NGC4472.fit), span=0.
05),
   lwd=2)
```

We can perform more analysis of the residuals. First, we show the residuals are normally distributed (Shapiro-Wilks test) but exhibit strong spatial autocorrelation (Durbin-Watson test).

In [ ]:
```
# Test for normality (OK) and autocorrelation (not OK) of residuals
# For linear models, also use the Durbin-Watson test in CRAN packages
lmtest and car

qqnorm(residuals(NGC4472.fit) / summary(NGC4472.fit)$sigma)
abline(a=0,b=1)
shapiro.test(residuals(NGC4472.fit) / summary(NGC4472.fit)$sigma)
acf(residuals(NGC4472.fit))
```

There is an oddity: the error on Sersic's `n` parameter from `nls` is much smaller than the error quoted by Kormendy. So we check this with a bootstrap analysis, and confirm that `nls` is correct. Reading Kormendy's appendix, we find that he did not know how to evaluate the uncertainty of a nonlinear fit and chose an ad hoc procedure that was inaccurate.

In [ ]:
```
# Bootstrap parameter estimates
# Note bootstrap here does not account for autocorrelation in the resi
duals
# Both the nls and the bootstrap error on Sersic parameter 'n' are muc
h smaller
# than that estimated by Kormendy et al. (2009) using heuristic method
s.

install.packages('nlstools')  ;  library(nlstools)
NGC4472.boot <- nlsBoot(NGC4472.fit)
coef(NGC4472.fit) ; confint(NGC4472.fit)        # analytic fitted para
meters errors
summary(NGC4472.boot)                                                  # boot
strap fitted parameter errors
hist(NGC4472.boot$coefboot[,3], breaks=50, xlab='Sersic n', main='Erro
r analysis for Sersic fit of NGC 4472')
curve(dnorm(x,m=5.95, sd=0.10)*58/5.95, xlim=c(5.6,6.4), ylim=c(0,50),
add=T)

# Parameter ellipse with bootstrap distribution

install.packages('ellipse') ; library(ellipse)
plot(ellipse(NGC4472.fit, which=c(2,3), level=0.997), type='l', lwd=2)
lines(ellipse(NGC4472.fit, which=c(2,3), level=0.95), type='l', lwd=2)
lines(ellipse(NGC4472.fit, which=c(2,3), level=0.67), type='l', lwd=2)
points(NGC4472.fit$m$getPars()[[2]], NGC4472.fit$m$getPars()[[3]], pch
=25, lwd=2)
points(NGC4472.boot$coefboot[,2:3], pch=20, col='#66004470', cex=0.3)
```