Statistics in Environmental Research (BUC Workshop Series) I
Problem sheet - SOLUTIONS

1. We fit two models with and without latitude and longitude:

```
> source("scotdat.txt")
> summary(glm(Y~X+offset(log(E)),data=z,family=quasipoisson()))
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.5423     0.1542  -3.517 0.000893 ***
X             7.3732     1.3208   5.583 7.89e-07 ***
(Dispersion parameter for quasipoisson family taken to be 4.917963)
> summary(glm(Y~X+lat+long+offset(log(E)),data=z,family=quasipoisson()))
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -25.03815    4.75530  -5.265 2.70e-06 ***
X             5.49561    1.21123   4.537 3.40e-05 ***
lat           0.43537    0.08298   5.247 2.89e-06 ***
long          0.02316    0.07894   0.293     0.77
(Dispersion parameter for quasipoisson family taken to be 3.109075)
```

   Including latitude leads to a reduction in the size of the coefficient associated with AFF, since there is a south-north increasing, gradient in risk, and in exposure. Note how the level of overdispersion drops in the model including latitude and longitude.

2. There are 1126 children in the original dataset, and if we remove children who have missing values on any of the variables we reduce the dataset by 232 to 894 (a 21% loss, which is something to worry about).

```
> x <- read.csv("jan31.csv")
> gender <- x[,8]; age <- x[,9]; smoke <- x[,28]; income <- x[,54]
> school <- x[,55]; parental <- x[,22]; exposure <- x[,60]; illness <- x[,21]
> x2 <- cbind(gender,age,illness,smoke,income,school,exposure,parental)
> x2[x2== -999] <- NA
> x2 <- na.omit(x2)
> gender <- factor(x2[,1]); age <- x2[,2]; illness <- x2[,3]
> illness2 <- rep(0,length(illness))
> illness2[illness==1] <- 0; illness2[illness==2] <- 0
> illness2[illness==3] <- 1; illness2[illness==4] <- 1
> smoke <- factor(x2[,4]); income <- factor(x2[,5])
> school <- factor(x2[,6]); exposure <- 10*x2[,7]/max(x2[,7])
> parental <- factor(x2[,8])
> mod2un <- glm(cbind(illness2,1-illness2)~exposure,
            family=binomial(link=logit))
> mod2 <- glm(cbind(illness2,1-illness2)
   ~exposure+school+gender+age+income+parental+smoke,family="binomial")
> summary(mod2un)
```

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.41161    0.16631 -14.500   <2e-16 ***
exposure     0.07494    0.05399   1.388    0.165
```

Of these 894 children, 85 had 5 or more respiratory illnesses. An unadjusted analysis (ie no confounders) gives the relative risk associated with a 1-unit increase in exposure (recall exposure is scaled between 0 and 10) as $\exp(0.0794) = 1.078$ but the result is not statistically significance.

Including the potential counfounders school, gender, age, income, parental smoking, and smoking, gives an estimate of $\exp(0.116) = 1.123$ (a 12% increase in the odds of 5 or more illnesses between two children whose exposure differs by 10 units) with an associated p-value of 0.064.

3. (a) We begin by calculating the reference probabilities (via internal standardization), in order to calculate expected numbers. The data here are quite extensive and so we can reliably estimate the 16 probabilities (for the 2 genders, 2 races and 4 age groups). The code is shown below:

```
nstrata <- 16; ncounty <- 88; q <- Yagg <- Nagg <- rep(0,nstrata);
for (j in 1:nstrata){
  indj <- j+seq(0,ncounty-1)*16
  Yagg[j] <- sum(ohio$deaths[indj]); Nagg[j] <- sum(ohio$popn[indj])
  q[j] <- Yagg[j]/Nagg[j]
}
```

If we plot these probabilities we see that for men the non-white mortality risks are greater at every age group, while for the women the pattern is less clear (the numbers here are small, and so the standard errors are larger).

The above can also be obtained by fitting a saturated model – the natural model here is Poisson:

$$Y_j \sim \text{Poisson}(N_j \times p_j)$$

where Y_j and N_j are the number of cases and the population in strata j, and p_j is the risk that we are estimating. This gives MLEs: $\widehat{p}_j = Y_j/N_j$, $j = 1, ..., 16$.

We first check that the rare disease assumption of the Poisson model is valid, by fitting an alternative binomial model:

$$Y_j \sim \text{Binomial}(N_j, p_j).$$

```
age <- factor(ohio$age[1:16]); gender <- factor(ohio$sex[1:16])
race <- factor(ohio$race[1:16])
inter <- glm(Yagg~offset(log(Nagg))+age*race*gender,family=poisson())
qmod <- inter$fitted/Nagg
interb <- glm(cbind(Yagg,Nagg-Yagg)~age*race*gender,family=binomial())
qmodb <- interb$fitted
```

The parameter estimates and fitted values (and hence reference values) are virtually identical indicating that the Poisson approximation is fine.

```
Yagg
 [1]  306   57 1025  172 1477  182  768  100  176   40  507   81  733   62  391
[16]   35
Nagg
 [1] 474393  52706 429617  48382 319387  29972 139050  11610 504198  64824
[11] 476170  54662 408229  38767 244965  19366
cbind(inter$coeff,interb$coeff,qmod,qmodb)
                                                      qmod        qmodb
(Intercept)          -7.34620627 -7.34556103 0.0006450348 0.0006450348
age8                  1.30800477  1.30974822 0.0010814708 0.0010814708
age9                  1.96981574  1.97380571 0.0023858460 0.0023858460
age10                 2.14740714  2.15230040 0.0035550411 0.0035550411
race1                 0.51677296  0.51720977 0.0046244838 0.0046244838
gender1              -0.61403406 -0.61433018 0.0060723342 0.0060723342
age8:race1           -0.11796010 -0.11722424 0.0055231931 0.0055231931
age9:race1           -0.24439463 -0.24337581 0.0086132644 0.0086132644
age10:race1          -0.07242572 -0.06975046 0.0003490692 0.0003490692
age8:gender1         -0.19278364 -0.19381092 0.0006170554 0.0006170554
age9:gender1         -0.33201328 -0.33455520 0.0010647458 0.0010647458
age10:gender1        -0.62732987 -0.63097484 0.0014818338 0.0014818338
race1:gender1         0.05291565  0.05274695 0.0017955608 0.0017955608
age8:race1:gender1   -0.12118419 -0.12177055 0.0015992984 0.0015992984
age9:race1:gender1   -0.44104637 -0.44252990 0.0015961464 0.0015961464
age10:race1:gender1  -0.37302599 -0.37575786 0.0018072911 0.0018072911
```

In situations with sparser data or more strata, we may wish to estimate the probabilities using models. We first demonstrate for the full model that includes all interactions, and so reproduces the above probabilities.

Aside: there are two uses of likelihood ratio statistics (or deviances). We can check the overall fit of a model by looking at the residual deviance – if the model (which specifies the null hypothesis) is appropriate then the residual deviance will not be in the tail of a χ^2 distribution (unless we are unlucky!) with degrees of freedom given by the total number of independent observations minus the number of estimated parameters.

Another use is to check for a simplification of a model, if a simple model is a valid simplification then the change in the deviance will be probably be a reasonable observation from a χ^2 distribution with degrees of freedom given by the difference in number of parameters.

There are some conditions under which the use of the χ^2 distribution are valid, an important one is that the sample size is large (relative to the number of estimated parameters). Here we should be fine.

To illustrate how we might obtain more reliable estimates of reference probabilities we experiment with fitting simpler models, beginning with the main effects model, which the following code shows is clearly inadequate, the 95% point of a χ^2 statistic on 10 degrees of freedom is 18.3, and here the residual deviance is 73.8.

```
main <- glm(Yagg~offset(log(Nagg))+age+race+gender,family=poisson())
```

```
summary(main)
Coefficients:
            Estimate Std. Error  z value Pr(>|z|)
(Intercept) -7.19641    0.04266 -168.687  < 2e-16 ***
age8         1.21613    0.04783   25.428  < 2e-16 ***
age9         1.80921    0.04622   39.143  < 2e-16 ***
age10        1.89913    0.05011   37.902  < 2e-16 ***
race1        0.31317    0.03950    7.928 2.22e-15 ***
gender1     -0.96324    0.02727  -35.326  < 2e-16 ***
    Null deviance: 3763.131  on 15  degrees of freedom
Residual deviance:   73.826  on 10  degrees of freedom
```

Further fitting reveals that the model with the $age \times gender + age \times race$ interactions gives residual degrees of freedom 9.5 on 4 degrees of freedom, which coincides with the 95% point of a χ^2_4. The reference probabilities are only slightly different under the two model, expected numbers and SMRs are virtually identical.

The $gender \times race$ interaction is of marginal importance – on the log scale the race effect at each age appears to be approximately constant for men and women.

The expected numbers are calculated via

```
Obs <- Exp <- rep(0,ncounty)
for (i in 1:88){
  indi <- (i-1)*16+seq(1,16)
  Obs[i] <- sum(ohio$deaths[indi])
  Exp[i] <- sum(ohio$popn[indi]*q)
}
SMR <- Obs/Exp
OhioMap(SMR,ncol=8,type="e",figmain="Ohio lung cancer SMRs'')
county <- factor(seq(1,88)) # confirm that SMRs are MLEs
smrmod <- glm(Obs~-1+county+offset(log(Exp)),family=poisson()) # note: -1
smr2 <- exp(smrmod$coeff) # the same as SMR
```

The above also gives an alternative method of calculating the SMRs (to give `smr2`) using a Poisson model.

If mapped, he SMRs illustrate a relatively tight range, between 0.36 and 1.54.

(b) The standard errors of the SMRs show some variability.

(c)

(d) The smoothed estimates display a much narrower range since $\widehat{\alpha} = 51.7$, giving a standard deviation of the random effects of 0.14.

After plotting the smoothed estimates versus the SMRs the attenuation towards 1 is apparent.

(e) The posterior standard deviations of the smoothed estimates are much smaller (and more constant) than the SMR counterparts, due to the large value of α.

(f) If we plot the posterior probabilities that the relative risk exceeds 1.2, the plot confirms that the relative risks are close to 1 for these data, the maximum probability is 0.47.

(g) We now turn to examining whether proportionality was appropriate. For illustration we calculate SMRs for men and women.

```
ObsM <- ObsF <- ExpM <- ExpF <- rep(0,ncounty); qM <- q[1:8]; qF <- q[9:16]
for (i in 1:88){
  indiM <- (i-1)*16+seq(1,8); indiF <- indiM + 8
  ObsM[i] <- sum(ohio$deaths[indiM])
  ExpM[i] <- sum(ohio$popn[indiM]*qM)
  ObsF[i] <- sum(ohio$deaths[indiF])
  ExpF[i] <- sum(ohio$popn[indiF]*qF)
}
SMRM <- ObsM/ExpM
SMRF <- ObsF/ExpF
```

We examine the fit of various models. Here we have 1408 rows and 6112 cases so we might doubt the asymptotics. Hence we examine the Monte Carlo distribution of the LR statistic. We test three hypotheses:

$$\begin{aligned} H_0 &: race \times age \times sex \\ H_A &: \text{saturated model} \end{aligned}$$

which tests whether the stratum only model provides an adequate fit to the data (if it does then we don't need area effects).

$$\begin{aligned} H_0 &: race \times age \times sex + county \\ H_A &: \text{saturated model} \end{aligned}$$

which tests whether the stratum and area model provides an adequate fit to the data (if it does then we don't need to worry about looking at area effects in different stratum, i.e. the proportionality assumption implicit in the SMR calculation is OK). Finally:

$$\begin{aligned} H_0 &: race \times age \times sex \\ H_A &: race \times age \times sex + county \end{aligned}$$

assesses whether the area effects are needed.

The LR statistics for these 3 tests have degrees of freedom 1365 (1408-27-16) (27 is the number of zeros), 1278 (1365-87) and 87 respectively, giving deviances 1205, 1027 and 178 with asymptotic p-values of 1, 1 and 0.

The Monte Carlo p-values are 0.01, 0.064 and 0 (from 1000 simulations) showing that we have strong evidence that the stratum only model is inadequate (and that the asymptotic distributions were way off, a well-known fact, see for example Venables and Ripley, Modern Applied Statistics with S-Plus), the $county + race \times age \times race$ model is only marginally OK, and we always reject the former for the latter.

For the Monte Carlo distributions of the deviances:

```
Test stats Observed:
 1205.152 1365 0.9992502
 1027.276 1278 1
 177.8759 87 3.32756e-08
```

Here we conclude that the stratum only model is inadequate, and adding county is definitely a good idea. The county and stratum model is borderline adequate (fitting a quasiPoisson model gives an overdispersion parameter of 1.06, showing that there is not a lot of excess-Poisson variability in these data.

4. The R code is given by

```
nsims <- 50; nareas <- 50; E <- runif(nareas,.1,50)
msesmr1 <- 0; msesmr2 <- 0; msesmr3 <- 0
mseeb1 <- 0; mseeb2 <- 0; mseeb3 <- 0
for (s in 1:nsims){
  theta1 <- rgamma(nareas,5,5)
  theta2 <- rlnorm(nareas,0,1)
  theta3 <- runif(nareas,.1,10)
  y1 <- rpois(nareas,E*theta1)
  y2 <- rpois(nareas,E*theta2)
  y3 <- rpois(nareas,E*theta3)
  smr1 <- y1/E; smr2 <- y2/E; smr3 <- y3/E;
  msesmr1 <- msesmr1 + sum((smr1-theta1)^2)
  msesmr2 <- msesmr2 + sum((smr2-theta2)^2)
  msesmr3 <- msesmr3 + sum((smr3-theta3)^2)
  EBest1 <- eBayes(y1,E)
  EBest2 <- eBayes(y2,E)
  EBest3 <- eBayes(y3,E)
  mseeb1 <- mseeb1 + sum((EBest1$RR-theta1)^2)
  mseeb2 <- mseeb2 + sum((EBest2$RR-theta2)^2)
  mseeb3 <- mseeb3 + sum((EBest3$RR-theta3)^2)
}
cat("MSE Gamma data SMR EB :",msesmr1/(nsims*nareas),mseeb1/(nsims*nareas),"\n")
cat("MSE lognormal data SMR EB :",msesmr2/(nsims*nareas),mseeb2/(nsims*nareas),"\n")
cat("MSE uniform data SMR EB :",msesmr3/(nsims*nareas),mseeb3/(nsims*nareas),"\n")
```

Below we see that the empirical Bayes estimates have the smallest MSE in each case; not surprisingly the biggest gains are when the distribution is truly gamma (the squared error of a typical estimate under the EB model is about a third of that under the SMR in this case).

```
MSE Gamma data SMR EB : 0.1430656 0.04497075
MSE lognormal data SMR EB : 0.2421748 0.1684692
MSE uniform data SMR EB : 0.7616984 0.444556
```

The MSE is the variance plus the bias squared and so we see that the varaince of the SMRs is what is dominating (they are unbiased, while the EB estimates are biased towards 1).

We multiply the expected numbers by 10 and obtain the results:

```
MSE Gamma data SMR EB : 0.007316697 0.006517499
MSE lognormal data SMR EB : 0.01173368 0.0115257
MSE uniform data SMR EB : 0.03691264 0.03619011
```

showing that as we obtain more data the smoothed estimates do not differ greatly from the SMRs.

5. The code to simulate the case-control data is given below – the at risk population was taken to be the controls from the South Lancashire data.

```
# Set up distances
dist <- sqrt((southlancs$x[southlancs.cc==0]-old.incinerator[1])^2+
             (southlancs$y[southlancs.cc==0]-old.incinerator[2])^2)
pointmap(southlancs.pts[southlancs.cc==0,])
u <- runif(length(southlancs$x[southlancs.cc==0]))
beta0 <- 0; beta1 <- -log(300)/max(dist); simcc <- rep(0,length(u))
prob <- exp(beta0+beta1*(dist-mean(dist)))/(1+exp(beta0+beta1*(dist-mean(dist))))
# Let's plot the probability surface
xplot <- seq(min(southlancs$x[southlancs.cc==0]),max(southlancs$x[southlancs.cc==0]),1000)
yplot <- seq(min(southlancs$y[southlancs.cc==0]),max(southlancs$y[southlancs.cc==0]),1000)
distplot <- sqrt(outer((xplot-old.incinerator[1])^2,(yplot-old.incinerator[2])^2,"+"))
probplot <- exp(beta0+beta1*(distplot-mean(distplot)))/
            (1+exp(beta0+beta1*(distplot-mean(distplot))))
postscript("ex4q2_hv1000.ps",horiz=F)
par(mfrow=c(3,2))
# Plot 1
image(xplot,yplot,probplot)
# the next line assigns cases and controls to the population at risk
simcc <- ifelse (u<prob,1,0)
sim0 <- as.points(southlancs$x[simcc==0],southlancs$y[simcc==0])
sim1 <- as.points(southlancs$x[simcc==1],southlancs$y[simcc==1])
npts(sim0); npts(sim1); sim.bdy <- southlancs.bdy
# Plot 2
plot(sim0[,1],sim0[,2],xlab="Eastings (m)",ylab="Northings (m)",pch=3,cex=.5)
title("Control locations")
points(old.incinerator[1],old.incinerator[2],cex=2,pch=2)
# Plot 3
plot(sim1[,1],sim1[,2],xlab="Eastings (m)",ylab="Northings (m)",pch=3,cex=.5)
title("Case locations")
points(old.incinerator[1],old.incinerator[2],cex=2,pch=2)
hv <- 1000; ngrid <- 40
# Plot 4
polymap(sim.bdy,border="grey",xlab="Eastings (m)",ylab="Northings (m)")
points(sim0,pch=3,cex=.5)
contour(kernel2d(sim0,sim.bdy, h=hv, nx=ngrid, ny=ngrid), nlevels=10,add=T,drawlabels=F)
# Plot 5
polymap(sim.bdy,border="grey",xlab="Eastings (m)",ylab="Northings (m)")
points(sim1,pch=3,cex=.5)
contour(kernel2d(sim1,sim.bdy, h=hv, nx=ngrid, ny=ngrid), nlevels=10,add=T,drawlabels=F)
```

```
# Plot 6
polymap(sim.bdy,border="grey",xlab="Eastings (m)",ylab="Northings (m)")
contour(kernrat(sim1,sim0,sim.bdy,h1=hv,h2=hv), nlevels=20,add=T,drawlabels=T)
dev.off()
```

We assume that the risk at location $\boldsymbol{s}_i$ for individual i is

$$\text{risk} = \frac{\exp(\beta_0 + \beta_1[d_i - \overline{d}])}{1 + \exp(\beta_0 + \beta_1[d_i - \overline{d}])}$$

so that at distances from the source (the incinerator), d, the risk is:

$$\begin{aligned} d &= 0, \quad \text{risk} = \frac{\mathrm{e}^{\beta_0 - \beta_1 \overline{d}}}{1 + \mathrm{e}^{\beta_0 - \beta_1 \overline{d}}} \\ d &= \overline{d}, \quad \text{risk} = \frac{\mathrm{e}^{\beta_0}}{1 + \mathrm{e}^{\beta_0}} \\ d &= d_{\max}, \quad \text{risk} = \frac{\mathrm{e}^{\beta_0 + \beta_1[d_{\max} - \overline{d}]}}{1 + \mathrm{e}^{\beta_0 + \beta_1[d_{\max} - \overline{d}]}} \end{aligned}$$

which for the choices $\beta_0 = 0$, $\beta_1 = \log 300/d_{\max}$ give risks 0.96, 0.5 and 0.07. The choice of $\beta_0 = 0$ gives (on average) equal numbers of cases and controls.

The reconstruction clearly shows the risk close to the incinerato – but I had to choose a very large band width.

6. The data set `ca20` contains the calcium content measured in soil samples taken from the 0-20cm layer at 178 locations within a certain study area divided in three sub-areas. The elevation at each location was also recorded.

   The first region is tipically flooded during the rain season and not used as an experimental area. The calcium levels would represent the natural content in the region. The second region has received fertilizers a while ago and is typically occupied by rice fields. The third region has recieved fertilizers recently and is frequently used as an experimental area.

   On the basis of some exploratory plots of the data, and some preliminary least squares fits, we decide to include `area` and `Y coord` as regressors.

```
library(geoR)
data(ca20)
postscript("ex5q1_fig1.ps",horiz=F)
points(ca20)
mod1 <- lm(ca20$data~ca20$coords[,2]+ca20$cov$area)
summary(mod1)
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)      152.400781  36.200188   4.210 4.09e-05 ***
ca20$coords[, 2]  -0.020921   0.006491  -3.223  0.00151 **
ca20$cov$area2     6.989041   3.026845   2.309  0.02212 *
ca20$cov$area3     8.781387   3.969072   2.212  0.02824 *
Residual standard error: 9.392 on 174 degrees of freedom
Multiple R-Squared: 0.2937,     Adjusted R-squared: 0.2815
F-statistic: 24.12 on 3 and 174 DF,  p-value: 4.220e-13
```

The exponential correlation model is written as $\exp(-d/\phi)$ in `geoR`. (Note the difference with `GeoBUGSS`). Hence the distance at which the correlations fall to a half is

$$d_{0.5} = \log 2\phi \approx 0.7 \times \phi$$

and the correlations fall to 0.05 at

$$d_{0.05} = \log 20\phi \approx 3 \times \phi$$

The correlations fall to 0.36 (=1/e) at ϕ.

The nugget effect is σ_v^2.

```
par(mfrow=c(2,3))
plot(bin1)
lines.variomodel(cov.model="exp",cov.pars=c(50,200),nugget=0,max.dist=1000)
abline(v=log(2)*200); abline(h=50); abline(h=0)
plot(bin1)
lines.variomodel(cov.model="exp",cov.pars=c(50,600),nugget=0,max.dist=1000)
abline(v=log(2)*600); abline(h=50); abline(h=0)
plot(bin1)
lines.variomodel(cov.model="exp",cov.pars=c(100,200),nugget=0,max.dist=1000)
abline(v=log(2)*200); abline(h=100); abline(h=0)
plot(bin1)
lines.variomodel(cov.model="exp",cov.pars=c(100,600),nugget=0,max.dist=1000)
abline(v=log(2)*600); abline(h=100); abline(h=0)
plot(bin1)
lines.variomodel(cov.model="exp",cov.pars=c(100,200),nugget=50,max.dist=1000)
abline(v=log(2)*200); abline(h=150); abline(h=50)
plot(bin1,ylim=c(0,250))
lines.variomodel(cov.model="exp",cov.pars=c(200,500),nugget=50,max.dist=1000)
dev.off()

plot(bin1)
olsfit <- variofit(bin1,ini=c(150,500),weights="equal")
wlsfit <- variofit(bin1,ini=c(150,500))
lfit <- likfit(ca20,ini=c(150,280))
lfit2 <- likfit(ca20,ini=c(150,280),method=''REML'')
lines(olsfit,lty=1)
lines(wlsfit,lty=2)
lines(lfit,lty=3)
legend(500,50,legend=c("OLS","WLS","MLE"),lty=c(1,2,3),bty="n")
```

7. An AR(1) process is a model for time series, and is given by the following difference equation

$$X_t = \alpha_1 X_{t-1} + Z_t,$$

where $Z_t \sim N(0, \sigma_Z^2)$. That is, the output at time t depends linearly on the previous value of the process and on some noise term.

To simulate an AR(1) process with $n = 50$ and $\alpha_1 = 0.5$ we use the following R code.

```
sim <- arima.sim(n=50,model=list(ar=c(0.5)))
```

Note, that the processes is stationary only for $\alpha_1 \in (-1, 1)$.

When α_1 is close to 0, the output looks like random noise, the dependence on the previous value is not obvious from the plots. For larger values of α_1, as the correlation between X_t and X_{t-1} increases, the contribution of the previous value of the series becomes more visible. However the long-term fluctuation around the mean value might not be trivial when one is plotting only 50 elements of the series.

8. We start by extracting the relevant data from the dataset and plotting it by using the following code.

```
LabOzone <- read.table("LabOzone.txt")
site9 <- ts(LabOzone[,14])
plot(site9)
```

As a first step we can try to fit an `ARMA` model, that is we assume stationarity. The `ACF` and `PACF` plots suggest that the process has both `AR` and `MA` component, and the parameter of the autoregressive process is about 2. After trying a few variations of the model, we can see that the `ARMA(3,3)` is the best fit based on the AIC values. The model can be fitted by using the following `R` code.

```
fit <- arima(site9,order=c(3,0,3))
```

Then the model can be checked by using the code

```
tsdiag(fit)
```

which plots the standardised residuals, the `ACF` of the residuals and also the p-values for a Ljung-Box test. The plot of the `ACF` of the residuals show that there are still some significant correlations around lag 24.

To predict 24 hours ahead, we can use

```
pred1 <- predict(fit,n.ahead=24)
```

and then plot the prediction by using the following code.

```
plot(pred1$pred)
```

Based on the diagnostics (`ACF` of residuals) and the prediction, this model is not a good enough fit.

We can try to capture the significant correlation around lag 24 by assuming non-stationarity, which is a reasonable guess, as the plotted data doesn't seem the have a constant mean. After differencing and plotting the differenced process by using the codes

```
diff <- diff(site9,lag=1,difference=1)
plot(diff)
```

we see that now it looks stationary. After trying a few models, we find that the `ARIMA(1,1,1)` is the best fit, however running the diagnostics still reveals some significant correlations around lag 24, and also the prediction does not look too realistic.

Finally we can try to explain the correlation by seasonality, and fit a `SARIMA` model with a period of 24 hours. After playing around with the parameters, we find that a `SARIMA(1,0,1)(1,1,1)` gives the lowest AIC value. This model can be fitted by using the following `R` code

```
fit <- arima(site9,order=c(1,0,1),
             seasonal=list(order=c(1,1,1),period=24))
```

After fitting this model the significant spikes disappear on the plot of the `ACF` of residuals. The plot of the 24 hour prediction has a similar shape to the one we would expect based on the data. To have a better idea how the predicted values behave, we can also use the `forecast` function of the `forecast` package, and plot the resulting series.

9. The `sp` package is needed to access the `meuse` dataset.

   (a) When calculating the empirical variogram, the formula `log(zinc)~1` means that we assume constant trend for `log(zinc)`. For a theoretical variogram model we can use, among others, exponential, spherical, Gaussian or Matern class of models. A good inital guess could be 0.5 for the nugget, 0.6 for the sill and 800 for the range, thus we can use the code

   ```
   meuse.vfit <- fit.variogram(vgm1, vgm(0.5,"Mat",800,0.6))
   ```

   and then we can get the SSE associated with the model by using the following code.

   ```
   attr(meuse.vfit, 'SSErr')
   ```

   Based on the SSE criteria, the sperical model with SSE $= 9.011195e-06$ is the best fitted model for the variable `log(zinc)`.

(b) To do leave-one-out cross validation we need to set the number of folds to the number of samples we have, thus `nfold=length(meuse$zinc)`. The error can be obtained by using `x$residual`, while we get the variance using `x$var1.var`. For example, for the mean squared error we can use the following function.

```
MSE <- function(xv.obj){ tmp <- xv.obj$residual
                         return(sum(tmp^2)/length(tmp))
}
```

After calculating the MSE and MSDR, we still get, that the spherical model is the best in this case.