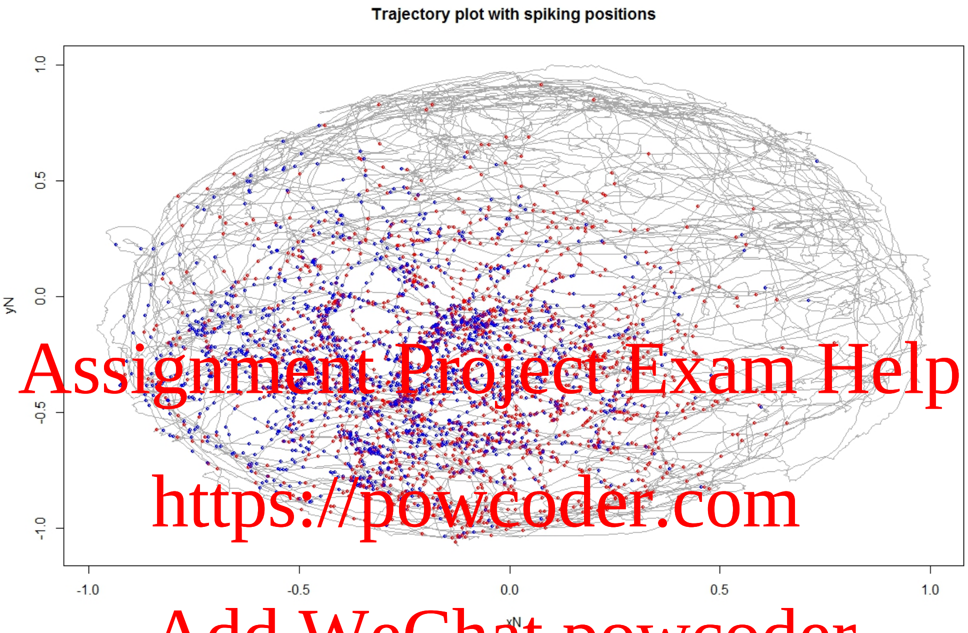
MA 568 Statistical Analysis of Point Process Data
Problem set 3 solution

Long Tao and Uri Eden
Department of Mathematics and Statistics, Boston university

Question 1

We download the *hipp_data.mat* file, which consists of the spiking activity of two neurons recorded simultaneously from the CA1 regions of a rat's hippocampus as it performed a free-foraging task. We plot the animal's movement trajectory with its position at the spiking times for each neuron overlaid.



From the plot above, we find that these two neurons are both tuned to the spatial location of the animal, and both have similar tuning properties in that their place fields nearly completely overlap. The center of both place fields occur when both the x and y coordinates are slightly negative.

```
spike_data <-read.csv("hipp_data.csv",header=T)
attach(spike_data)
#1.a Plot movements trajectory with position at each spiking times
plot(xN,yN,cex=.2,col='darkgray',type='l')
idx_1 = spikes>0; idx_2 = spikes2>0
points(xN[idx_1],yN[idx_1],col='red',cex=.5)
points(xN[idx_2],yN[idx_2],col='blue',cex=.5)
title('Trajectory plot with spiking positions')
```

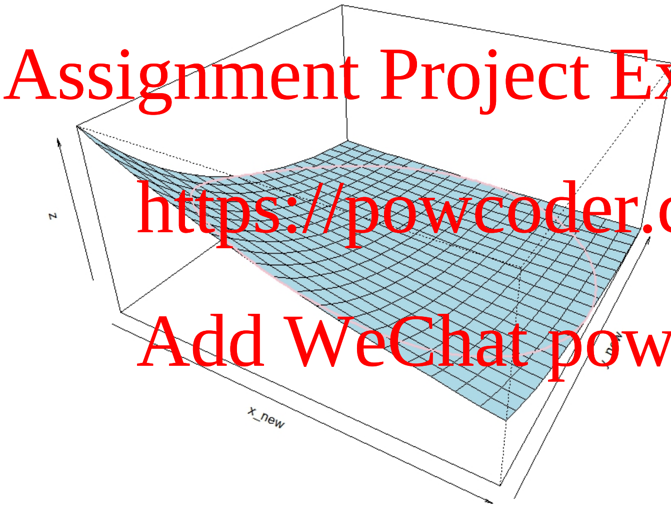
Question 2

Use glm command to fit the exponential linear model of equation

$$\lambda(t) = \exp\{\beta_0 + \beta_1 x_N(t) + \beta_2 y_N(t)\} \tag{1}$$

From the summary, we see all of the p values are highly significant ($p < 2e - 16$). We plot out the likelihood model intensity as a function of (x, y) position. The exponential linear model captures the fact that the firing rate increases as we move from the origin to negative values for x and y. However, the firing rate continues to increase in this direction even though the number of spikes falls off as x and y become more negative. In order to capture this phenomenon, we will need to include quadratic terms of x and y.

```
model1 <- glm(spikes~xN+yN, poisson , data)
summary(model1)
 Call:
glm(formula = spikes ~ xN + yN, family = poisson , data = data)
 Coefficients:
 Estimate Std. Error z value Pr(>|z|)
 (Intercept) -3.24918     0.02760 -117.74    <2e-16 ***
xN           -0.50623     0.04724  -10.72    <2e-16 ***
yN           -1.18010     0.04477  -26.36    <2e-16 ***
```



```
# construct a grid of positions to plot the model against...
x_new <- seq(-1,1,.1); y_new <- seq(-1,1,.1)

# compute lambda for each point on this grid using the GLM model
lambda <- function(x,y) { exp(predict(model1, data.frame(xN =x, yN=y))) }
z<- outer(x_new,y_new,lambda)

# plot lambda as a function position over this grid
persp(x_new, y_new, z, theta = 30, phi = 30, expand = 0.5, col = "lightblue") -> res
phi <- seq(0, 2*pi, len = 201)
xr <- cos(phi)
yr <- sin(phi)
lines(trans3d(xr,yr,lambda(xr,yr),res), col = "pink", lwd = 2)
```

Question 3

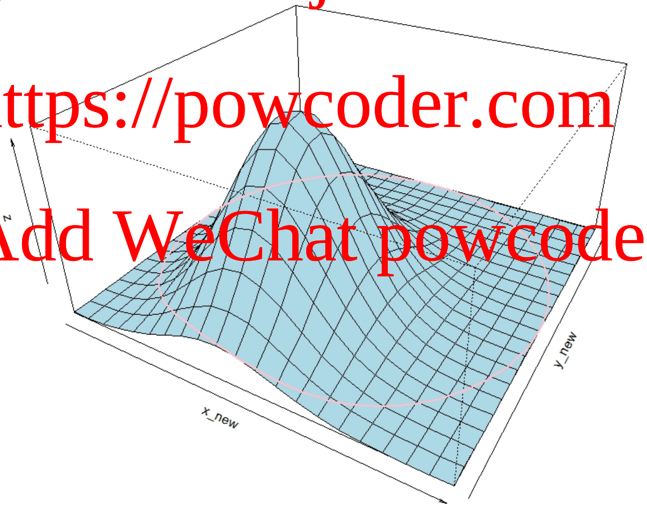
One way to improve this exponential linear model is to add quadratic functions of x_N and y_N. That is

$$\lambda(t) = \exp\{\beta_0 + \beta_1 x_N(t) + \beta_2 y_N(t) + \beta_3 x_N(t)^2 + \beta_4 y_N(t)^2 + \beta_5 x_N : yN\} \tag{2}$$

all of the parameters except the one corresponding to the x_N : y_N term are significant. This suggests that a more parsimonious model could be constructed by removing this term from the model.

The AIC-values for exponential linear and quadratic model are 11394.2 and 9153.2 respectively.

Actually here we are fitting the Gaussian place field model, and it is able to more accurately describe the firing properties of each of these neurons. Each place field has a center where firing rate is maximal, and falls off evenly when in any direction away. Even with the extraneous x_N : y_N term, the Gaussian model has significantly lower AIC value.



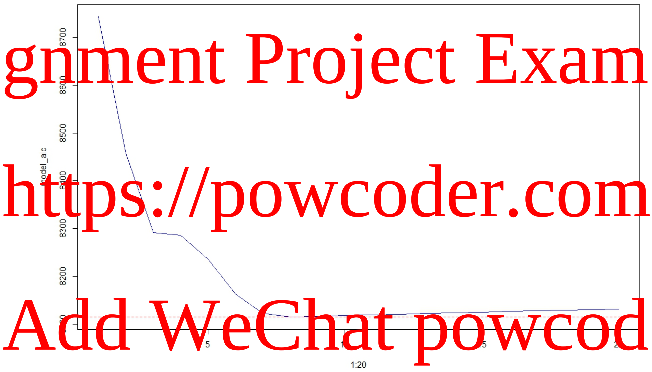
```
# fit a GLM model to the x and y positions.  Adjust this line to improve model fit.
model2 <- glm(spikes~xN*yN + I(xN^2) + I(yN^2),poisson,data)
summary(model2)
# visualize your model
# construct a grid of positions to plot the model against...
x_new <- seq(-1,1,.1)
y_new <- seq(-1,1,.1)

# compute lambda for each point on this grid using the GLM model
```

```
lambda <- function(x,y) { exp(predict(model2, data.frame(xN=x,yN=y))) }
z<- outer(x_new,y_new,lambda)
# plot lambda as a function position over this grid
persp(x_new, y_new, z, theta = 30, phi = 30, expand = 0.5, col = "lightblue") -> res
phi <- seq(0, 2*pi, len = 201)
xr <- cos(phi)
yr <- sin(phi)
lines(trans3d(xr,yr,lambda(xr,yr),res), col = "pink", lwd = 2)
# Compute the AIC
model2$deviance +2*6    # 9153.217
model1$deviance +2*3    # 11394.16
```

Question 4

The AIC is minimized by a history component that goes back 8 ms. The optimal model from this model class would include the constant term, the 4 quadratic terms with significant parameters and 8 history terms.



```
library(quantmod)
spike_hist = Lag(spikes, k=1:20)
n=length(T)
#spike_hist[1:20,1:20] = data[1:20,5:24]
idx=21:n
for (p in 1:20) {
model <- glm(spikes[idx]~xN[idx]*yN[idx] + I(xN[idx]^2) + I(yN[idx]^2) + spike_hist[idx,1:p],
family = poisson,data=data)
model_aic[p] = model$deviance + 2*length(model$coefficients)
}
plot(1:20,model_aic, type='l',col='blue')
abline(h=min(model_aic),lty=2,col='red')
which.min(model_aic) # 8
```

Question 5

Augment the GLM model from question 4 to include the past spiking history of the other recorded neuron. And we find that none of the parameter estimates related to the history of the second neuron is significant. This suggests that there is no functional interaction from neuron 2 to neuron

1, once position effects have been modeled. The optiomal model remains the one with a constant term, the 4 quadratic terms with significant parameters, 8 history terms and no interaction terms.

```
spike_hist = Lag(spikes, k=1:20)
n=length(T)
spike_hist2 = Lag(spikes2, k = 1:20)
model <- glm(spikes[idx]~xN[idx]*yN[idx] + I(xN[idx]^2) + I(yN[idx]^2) +
spike_hist[idx,1:8] + spike_hist2[idx,],family = poisson,data=data)
summary(model)
```

Question 6

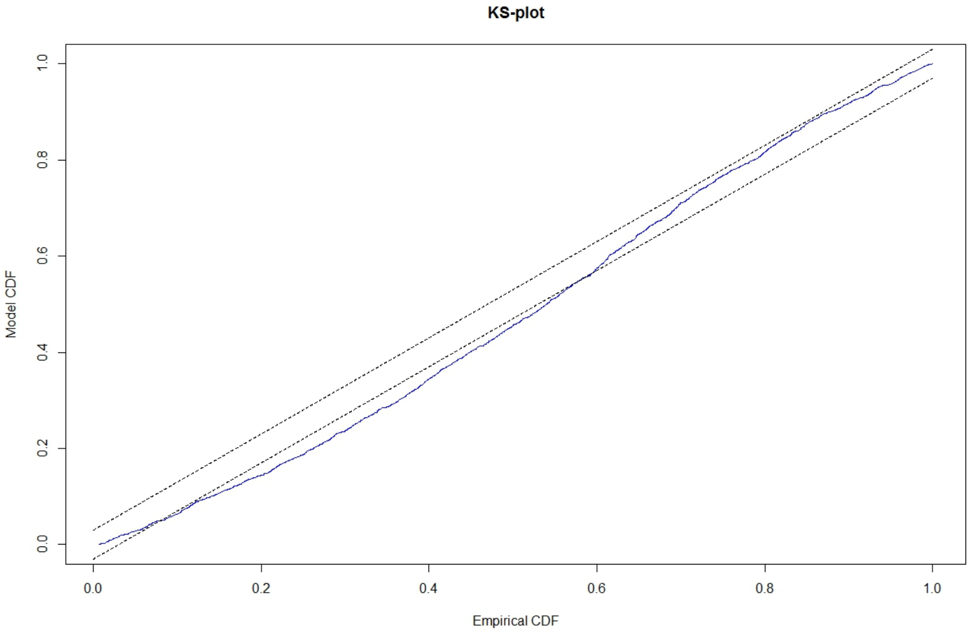
When we eliminate the position related terms from the model, all of the interaction parameters become significant. This is not surprising, since any spiking neuron 2 is likely to correspond to periods when the animal is near the center of place field 2, and therefore, near the center of the palce field of neuron 1. This model uses information about position that is important in the firing of neuron 2 to predict the spiking of neuron 1. When we add the position term, this information is already present and no significant interaction terms are observed.

```
idx=21:n
model <- glm(spikes[idx]~ I(xN[idx]^2) + I(yN[idx]^2) +
spike_hist[idx,1:8] + spike_hist2[idx,],family = poisson,data=data)
summary(model)
```

Question 7

The most parsimonious model does not completely pass the KS test. This suggests that there still may be other statistical structure in the spiking activity that cannot be captured within this model class. However, the KS statistic is relatively small suggesting that this model is able to explain most of the observed structure in the data.

```
model <- glm(spikes~xN+yN+I(xN^2)+I(yN^2)+
spikes_hist1+spikes_hist2+spikes_hist3+spikes_hist4+spikes_hist5+
spikes_hist6+spikes_hist7+spikes_hist8,family = poisson,data=data)
summary(model)
lambda_hat = exp(predict(model)); summary(lambda_hat)
lambda_hat = model$fitted.values; summary(lambda_hat)
lambda_int = cumsum(lambda_hat)
Z = diff(c(0,lambda_int[spikes>0]))
U = pexp(Z)
m=length(U)
plot(sort(U), (1:m-1)/m,type='l',lwd=1,col='blue',xlab='Empirical CDF',
ylab='Model CDF')
lines((1:m-1)/m,(1:m-1)/m+1.36/sqrt(m),lty=2)
lines((1:m-1)/m,(1:m-1)/m-1.36/sqrt(m),lty=2)
title('KS-plot')
```


KS-plot
Model CDF
Empirical CDF
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0

c