**Summary**

This study aims to predict the weight on any day between conception and birth of a pregnant woman. An experiment lasting for approximately 60 weeks was conducted. Weights (Ibs) of a pregnant woman on each day and the specific hour of the day (24-hour clock) since last menstrual period were observed. Based on this study, 284 observed data with 2 outliers being discarded were selected from the given dataset.

Using a piecewise linear function, days between conception and birth are separated into two and three parts respectively. Normal probability models with four different means depending on days in two or three periods, with and without mean weights also depending on the hour, were fitted to the data. A normal model with mean depending on days in three periods was chosen based on AIC and diagnostic tests. Diagnostic tests suggested that the key assumptions of this model had been met.

**The Data**

The data consist of 153 observed weights ranging from the day since last menstrual period to the day the pregnant woman gave birth, with associated hours, and 133 unknown weights with associated days and hours. They are shown in Figure 1. The figure suggests that the last two observed weights are from the data after birth which should not be included in this study. Furthermore, it suggests a reduction in weights in first trimester (day 0-day 91), and an increase in weights in second trimester (day 92-day 189) and in third trimester (day 190-day 280). Two outliers are discarded on conducting the models. The third trimester is then from day 190 to day 278.

**The Models**

Based on the figure 1, a normal distribution would be plausible. We treat the data as observations of identical, independently distributed (iid) normal random variables (Yi for the day i). The mean of the normal distribution is considered as a function of the days or a function of the days and associated hours. **y** is a vector containing the 151 observed weights.

Model 1:

I separate the days into three parts -first, second and third trimester.

μ=β0+β1x1+β2x2+β3x3. **θ**=(β0, β1, β2, β3, σ2) is the unknown parameter vector.

**x** is the day: 1= “first trimester”, 2= “second trimester”, 3= “third trimester”.

Since the mean weights given days is represented by connected line segments, I created a separate “dummy variable” for days. That is, define variables xi1, xi2, xi3 such that xi1 is the sequence of (0,…,91) if the associated days are from first trimester and xi1=91 otherwise. xi2 is the sequence of (1,…,98) if it is second trimester, xi2=0 if it is first trimester and xi2=98 if it is third trimester. xi3 is the sequence of (1,…,89) if it is third trimester and xi3=0 otherwise.

The likelihood is L(**θ; y, x**) =

Model 2:

Based on the figure 1, it seems that second and third trimester could be combined as one part.

μ=β0+β1x1+β2x2. **θ**=(β0, β1, β2, ). **x** are the days: 1= “the first trimester”, 2= “the second and third trimester”.

“dummy variable” in this case is: define variables xi1, xi2 such that xi1 is the sequence of (0,…,91) if it is first trimester and xi1=91 otherwise. xi2 is the sequence of (1,…,278) if it is second and third trimester and xi2=0 otherwise.

The likelihood is L(**θ; y, x**) =

Model 3:

Based on Model 1, I want to check if the mean weights also depend on the hour.

μ=β0+β1x1+β2x2+β3x3+ β4h. **θ**=(β0, β1, β2, β3, β4, σ2).

**x** is the day: 1=“first trimester”, 2=“second trimester”, 3= “third trimester”.

**h** is thespecifichour associated with the day.

The likelihood is L(**θ; y, x, h**) =

Model 4:

Based on Model 2, I want to check if the mean weights also depends on the hour.

μ=β0+β1x1+β2x2+β3h. **θ**=(β0, β1, β2, β3, σ2). **x** is the day: 1= “first trimester”, 2= “second trimester”. **h** is thespecifichour associated with the day.

The likelihood is L(**θ; y, x, h**) =

A close up of a map

Description automatically generated

Figure 1: Figure showing the day between conception and birth of a pregnant woman with various weights

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Mean | Parameter |  |
| Model 1 | μ=β0+β1x1+β2x2+β3x3 | **θ**=(β0, β1, β2, β3, σ2) | 5.3251 |
| Model 2 | μ=β0+β1x1+β2x2 | **θ**=(β0, β1, β2, ) | 24.5989 |
| Model 3 | μ=β0+β1x1+β2x2+β3x3+ β4h | **θ**=(β0, β1, β2, β3, β4, σ2) | 0 |
| Model 4 | μ=β0+β1x1+β2x2+β3h | **θ**=(β0, β1, β2, β3, σ2) | 21.2524 |

Table 1: Models and associated (the difference between AIC for the given model and the minimum AIC)

**Results and Conclusions**

The models were fitted by maximum likelihood and parameter variance-covariance matrices were estimated by the inverse Hessian obtained in the maximization process. I initially selected the model based on AIC and LRT. Model 3 where μ=β0+β1x1+β2x2+β3h has the lowest AIC among the four models (see Table 1). LRT test was constructed for Model 1 and 3 to decide whether the hour parameter could be added. The p-value is 0.021(3dp), and we would reject the null hypothesis in the 5% level that β3=0. It seems that Model 3 is preferable. However, diagnostic tests suggested that this model’s assumption of normality seems to be violated. Thus, I chose Model 1 with the second-lowest AIC.

Diagnostic Check for Model 1

The assumption of independent weights across the days was tested using a runs test on the residuals: yi-muhat(**x**), where muhat=β0+β1x1+β2x2+β3x3 and “hat” indicates maximum likelihood estimates of parameters. The Residual Histogram is shown in Figure 2 (left). The runs test of the null hypothesis (H0) of independence of residuals gave a p-value of 1.7e-06 so that H0 was rejected. It is potential because of physical effect, for instance, the event of the woman doing exercise on a particular day may affect the weights in the future. The adequacy of the selected Model was tested using a Q-Q plot (Figure 2 (right)) and Kolmogorov-Smirnov (KS) goodness-of-fit test of the null hypothesis that the data come from the fitted model, against the alternate hypothesis that it did not. The Q-Q plot does not suggest very substantial deviation from the fitted model, and the KS p-value was found to be 0.06 so that H0 was not rejected. However, the distribution of residuals has a heavy right tail, which is also shown in the right of Q-Q plot, where some points are relatively far above the reference line. This suggests that this model could be potentially optimised by reconsidering the trimester boundary or the length of the data.

A picture containing computer, game

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Figure 2: Figure showing a Residual Histogram (left) and a Q-Q plot for Model 1 (right)

**Appendix:**

setwd("~/Desktop/MT3508 project")

data<-read.table("MT3508 project data.txt",header=T)

# check relationship between day and weights

plot(data$day,data$weights,pch=19,cex=0.8,xlab="Days",ylab="Weights")

# Model 1: expec(y)=beta0+beta1\*x1(first)+beta2\*x2(second)+beta3\*x3(third)

# AIC: 333.2507

data1=data[c(1:284),c(1:4)]

data1=na.omit(data1)

negllik1=function(trtheta,data1){

n=dim(data1)[1]

mu=trtheta[1]+trtheta[2]\*data1$first+trtheta[3]\*data1$second+trtheta[4]\*data1$third

sigma2=exp(trtheta[5])

ll=-n/2\*log(2\*pi)-n/2\*log(sigma2)-sum((data1$weights-mu)^2)/(2\*sigma2)

return(-ll)

}

start=c(rep(0,4),log(var(data1$weights)))

negllik1(start,data1)

mle1.opt=optim(start,negllik1,data1=data1,hessian=TRUE,method="BFGS")

mu=c(mle1.opt$par[1:4])

mu

[1] 136.83279729 -0.03712647 0.15555299 0.17664981

sigma2=exp(mle1.opt$par[5])

sigma2

[1] 0.5320047

# 95% CI using Hessian

H=mle1.opt$hessian

vcv=solve(H)

beta0.ci=mle1.opt$par[1]+c(-1.96,1.96)\*sqrt(vcv[1,1])

beta0.ci

[1] 136.2225 137.4431

beta1.ci=mle1.opt$par[2]+c(-1.96,1.96)\*sqrt(vcv[2,2])

beta1.ci

[1] -0.04635698 -0.02789597

beta2.ci=mle1.opt$par[3]+c(-1.96,1.96)\*sqrt(vcv[3,3])

beta2.ci

[1] 0.1508258 0.1602802

beta3.ci=mle1.opt$par[4]+c(-1.96,1.96)\*sqrt(vcv[4,4])

beta3.ci

[1] 0.1710829 0.1822167

logsigma2.ci=mle1.opt$par[5]+c(-1.96,1.96)\*sqrt(vcv[5,5])

exp(logsigma2.ci)

[1] 0.4245808 0.6666079

# AIC

AIC.1=2\*mle1.opt$value+2\*length(mle1.opt$pars)

AIC.1

[1] 333.2507

# Expected values

n=dim(data1) [1]

X=matrix(c(rep(1,n),data1$first,data1$second,data1$third),ncol=4)

head(X)

[,1] [,2] [,3] [,4]

[1,] 1 28 0 0

[2,] 1 30 0 0

[3,] 1 31 0 0

[4,] 1 32 0 0

[5,] 1 32 0 0

[6,] 1 34 0 0

E.y=X%\*%mle1.opt$par[1:4]

# Residuals

library(TSA)

resids=data1$weights-E.y

# Run test on residuals

runs(resids,0)$pvalue

[1] 1.7e-06

plot(density(resids))

# Goodness-of-fits (KS on residuals)

ks.test(resids,"pnorm",alternative="two.sided")

One-sample Kolmogorov-Smirnov test

data: resids

D = 0.10692, p-value = 0.06332

alternative hypothesis: two-sided

# residuals against fitted

pos=(resids)>0

col=rep("blue",length(resids))

col[pos]="red"

plot(E.y,resids,col=col,pch=19)

abline(0,0,lty=2)

# QQ-plot

qqnorm(resids)

qqline(resids,lty=2)

# glm for model 1

fit1=glm(weights~first+second+third,family=gaussian,data=data1)

summary(fit1)

anova(fit1,test="F")

1-pchisq(deviance(fit1),df.residual(fit1))

plot(residuals(fit1,type="response")~predict(fit1,type="response"),xlab=expression(hat(mu)),ylab="Raw residuals")

plot(residuals(fit1,type="deviance")~predict(fit1,type="response"),xlab=expression(hat(mu)),ylab="Deviance residuals")

plot(residuals(fit1,type="pearson")~predict(fit1,type="response"),xlab=expression(hat(mu)),ylab="Pearson residuals")

# lm for model 1

fit1lm=lm(weights~first+second+third,family=gaussian,data=data1)

summary(fit1lm)

ols\_plot\_resid\_qq(fit1lm)

ols\_test\_normality(fit1lm)

ols\_test\_correlation(fit1lm)

ols\_plot\_resid\_fit(fit1lm)

ols\_plot\_resid\_hist(fit1lm)

plot(fit1lm)

# Model 2: expec(y)=beta0+beta1\*x1(first)+beta23\*x23(second and third)

# AIC: 352.5245

data2<-read.table("MT3508 project data 2.txt",header=T)

data2=data2[c(1:284),c(1:3)]

data2=na.omit(data2)

negllik2=function(trtheta,data2){

n=dim(data2)[1]

mu=trtheta[1]+trtheta[2]\*data2$first+trtheta[3]\*data2$second

sigma2=exp(trtheta[4])

ll=-n/2\*log(2\*pi)-n/2\*log(sigma2)-sum((data2$weights-mu)^2)/(2\*sigma2)

return(-ll)

}

start=c(rep(0,3),log(var(data2$weights)))

negllik2(start,data2)

mle2.opt=optim(start,negllik2,data2=data2,hessian=TRUE,method="BFGS")

mu=c(mle2.opt$par[1:3])

mu

[1] 137.27823975 -0.04672614 0.16500378

sigma2=exp(mle2.opt$par[4])

sigma2

[1] 0.6045461

# 95% CI using Hessian

H=mle2.opt$hessian

vcv=solve(H)

beta0.ci=mle2.opt$par[1]+c(-1.96,1.96)\*sqrt(vcv[1,1])

beta0.ci

[1] 136.6611 137.8954

beta1.ci=mle2.opt$par[2]+c(-1.96,1.96)\*sqrt(vcv[2,2])

beta1.ci

[1] -0.05551253 -0.03793974

beta2.ci=mle2.opt$par[3]+c(-1.96,1.96)\*sqrt(vcv[3,3])

beta2.ci

[1] 0.1624701 0.1675375

logsigma2.ci=mle2.opt$par[4]+c(-1.96,1.96)\*sqrt(vcv[4,4])

exp(logsigma2.ci)

[1] 0.4824644 0.7575189

# AIC

AIC.2=2\*mle2.opt$value+2\*length(mle2.opt$pars)

AIC.2

[1] 352.5245

# Expected values

n=dim(data2) [1]

X=matrix(c(rep(1,n),data2$first,data2$second),ncol=3)

head(X)

[,1] [,2] [,3]

[1,] 1 28 0

[2,] 1 30 0

[3,] 1 31 0

[4,] 1 32 0

[5,] 1 32 0

[6,] 1 34 0

E.y=X%\*%mle2.opt$par[1:3]

# Residuals

library(TSA)

resids=data2$weights-E.y

# Run test

runs(resids,0)$pvalue

[1] 2.94e-07

plot(density(resids))

# Goodness-of-fits (KS on residuals)

ks.test(resids,"pnorm",alternative="two.sided")

One-sample Kolmogorov-Smirnov test

data: resids

D = 0.10206, p-value = 0.08608

alternative hypothesis: two-sided

# residuals against fitted

pos=(resids)>0

col=rep("blue",length(resids))

col[pos]="red"

plot(E.y,resids,col=col,pch=19)

abline(0,0,lty=2)

# QQ-plot

qqnorm(resids)

qqline(resids,lty=2)

# glm for model 2

fit2=glm(weights~first+second,family=gaussian,data=data2)

summary(fit2)

anova(fit2,test="F")

1-pchisq(deviance(fit2),df.residual(fit2))

plot(residuals(fit2,type="response")~predict(fit2,type="response"),xlab=expression(hat(mu)),ylab="Raw residuals")

plot(residuals(fit2,type="deviance")~predict(fit2,type="response"),xlab=expression(hat(mu)),ylab="Deviance residuals")

plot(residuals(fit2,type="pearson")~predict(fit2,type="response"),xlab=expression(hat(mu)),ylab="Pearson residuals")

# lm for model 2

fit2lm=lm(weights~first+second,family=gaussian,data=data2)

summary(fit2lm)

ols\_plot\_resid\_qq(fit2lm)

ols\_test\_normality(fit2lm)

ols\_test\_correlation(fit2lm)

ols\_plot\_resid\_fit(fit2lm)

ols\_plot\_resid\_hist(fit2lm)

plot(fit2lm)

# Model3: based on model1 but add hour

# AIC: 327.9256

data1hr=data[c(1:284),c(1:5)]

data1hr=na.omit(data1hr)

negllik1hr=function(trtheta,data1hr){

n=dim(data1hr)[1]

mu=trtheta[1]+trtheta[2]\*data1hr$first+trtheta[3]\*data1hr$second+trtheta[4]\*data1hr$third+

trtheta[5]\*data1hr$hours

sigma2=exp(trtheta[6])

ll=-n/2\*log(2\*pi)-n/2\*log(sigma2)-sum((data1hr$weights-mu)^2)/(2\*sigma2)

return(-ll)

}

start=c(rep(0,5),log(var(data1hr$weights)))

negllik1hr(start,data1hr)

mle1hr.opt=optim(start,negllik1hr,data1hr=data1hr,hessian=TRUE,method="BFGS")

mu=c(mle1hr.opt$par[1:5])

mu

[1] 138.08969345 -0.03753615 0.15510753 0.17701214 -0.20428381

sigma2=exp(mle1hr.opt$par[6])

sigma2

[1] 0.5136647

# 95% CI using Hessian

H=mle1hr.opt$hessian

vcv=solve(H)

beta0.ci=mle1hr.opt$par[1]+c(-1.96,1.96)\*sqrt(vcv[1,1])

beta0.ci

[1] 136.8727 139.3067

beta1.ci=mle1hr.opt$par[2]+c(-1.96,1.96)\*sqrt(vcv[2,2])

beta1.ci

[1] -0.04661315 -0.02845916

beta2.ci=mle1hr.opt$par[3]+c(-1.96,1.96)\*sqrt(vcv[3,3])

beta2.ci

[1] 0.1504476 0.1597675

beta3.ci=mle1hr.opt$par[4]+c(-1.96,1.96)\*sqrt(vcv[4,4])

beta3.ci

[1] 0.1715335 0.1824908

beta4.ci=mle1hr.opt$par[5]+c(-1.96,1.96)\*sqrt(vcv[5,5])

beta4.ci

[1] -0.37626807 -0.03229955

logsigma2.ci=mle1hr.opt$par[6]+c(-1.96,1.96)\*sqrt(vcv[6,6])

exp(logsigma2.ci)

[1] 0.4099356 0.6436411

# AIC

AIC.3=2\*mle1hr.opt$value+2\*length(mle1hr.opt$pars)

AIC.3

[1] 327.9256

# Expected values

n=dim(data1hr) [1]

X=matrix(c(rep(1,n),data1hr$first,data1hr$second,data1hr$third,data1hr$hours),ncol=5)

head(X)

[,1] [,2] [,3] [,4] [,5]

[1,] 1 28 0 0 5.850000

[2,] 1 30 0 0 6.733333

[3,] 1 31 0 0 5.800000

[4,] 1 32 0 0 5.450000

[5,] 1 32 0 0 5.900000

[6,] 1 34 0 0 5.900000

E.y=X%\*%mle1hr.opt$par[1:5]

# Residuals

library(TSA)

resids=data1hr$weights-E.y

# Run test

runs(resids,0)$pvalue

[1] 9.08e-06

plot(density(resids))

# Goodness-of-fits (KS on residuals)

ks.test(resids,"pnorm",alternative="two.sided")

One-sample Kolmogorov-Smirnov test

data: resids

D = 0.11411, p-value = 0.03918

alternative hypothesis: two-sided

# residuals against fitted

pos=(resids)>0

col=rep("blue",length(resids))

col[pos]="red"

plot(E.y,resids,col=col,pch=19)

abline(0,0,lty=2)

# QQ plot

qqnorm(resids)

qqline(resids)

# glm for model 3

fit3=glm(weights~first+second+third+hours,family=gaussian,data=data1hr)

summary(fit3)

1-pchisq(deviance(fit3),df.residual(fit3))

plot(residuals(fit3,type="response")~predict(fit3,type="response"),xlab=expression(hat(mu)),ylab="Raw residuals")

plot(residuals(fit3,type="deviance")~predict(fit3,type="response"),xlab=expression(hat(mu)),ylab="Deviance residuals")

plot(residuals(fit3,type="pearson")~predict(fit3,type="response"),xlab=expression(hat(mu)),ylab="Pearson residuals")

anova(fit3,test="F")

confint(fit3)

anova(fit1,fit3,test="F")

# lm for model 3

fit3lm=lm(weights~first+second+third+hours,data=data1hr)

summary(fit3lm)

ols\_plot\_resid\_qq(fit3lm)

ols\_test\_normality(fit3lm)

ols\_test\_correlation(fit3lm)

ols\_plot\_resid\_fit(fit3lm)

ols\_plot\_resid\_hist(fit3lm)

plot(fit3lm)

# LRT (model 1 and model 3)

llik1=-mle1.opt$value

llik2=-mle1hr.opt$value

LRT=2\*(llik2-llik1)

LRT

[1] 5.32515

pchisq(LRT,df=1,lower.tail=FALSE)

[1] 0.0210198

# Model4: based on model2 but add hour

# AIC: 349.178

data2hr<-read.table("MT3508 project data 2.txt",header=T)

data2hr=data2hr[c(1:284),c(1:4)]

data2hr=na.omit(data2hr)

negllik2hr=function(trtheta,data2hr){

n=dim(data2hr)[1]

mu=trtheta[1]+trtheta[2]\*data2hr$first+trtheta[3]\*data2hr$second+trtheta[4]\*data2hr$hours

sigma2=exp(trtheta[5])

ll=-n/2\*log(2\*pi)-n/2\*log(sigma2)-sum((data2hr$weights-mu)^2)/(2\*sigma2)

return(-ll)

}

start=c(rep(0,4),log(var(data2hr$weights)))

negllik2hr(start,data2hr)

mle2hr.opt=optim(start,negllik2hr,data2hr=data2hr,hessian=TRUE,method="BFGS")

mu=c(mle2hr.opt$par[1:4])

mu

[1] 138.35599715 -0.04739365 0.16493437 -0.17268901

sigma2=exp(mle2hr.opt$par[5])

sigma2

[1] 0.5912952

# 95% CI using Hessian

H=mle2hr.opt$hessian

vcv=solve(H)

beta0.ci=mle2hr.opt$par[1]+c(-1.96,1.96)\*sqrt(vcv[1,1])

beta0.ci

[1] 137.0555 139.6565

beta1.ci=mle2hr.opt$par[2]+c(-1.96,1.96)\*sqrt(vcv[2,2])

beta1.ci

[1] -0.05611228 -0.03867503

beta2.ci=mle2hr.opt$par[3]+c(-1.96,1.96)\*sqrt(vcv[3,3])

beta2.ci

[1] 0.1624275 0.1674412

beta3.ci=mle2hr.opt$par[4]+c(-1.96,1.96)\*sqrt(vcv[4,4])

beta3.ci

[1] -0.35668728 0.01130925

logsigma2.ci=mle2hr.opt$par[5]+c(-1.96,1.96)\*sqrt(vcv[5,5])

exp(logsigma2.ci)

[1] 0.4718895 0.7409151

# AIC

AIC.4=2\*mle2hr.opt$value+2\*length(mle2hr.opt$pars)

AIC.4

[1] 349.178

# Expected values

n=dim(data2hr) [1]

X=matrix(c(rep(1,n),data2hr$first,data2hr$second,data2hr$hours),ncol=4)

head(X)

[,1] [,2] [,3] [,4]

[1,] 1 28 0 5.850000

[2,] 1 30 0 6.733333

[3,] 1 31 0 5.800000

[4,] 1 32 0 5.450000

[5,] 1 32 0 5.900000

[6,] 1 34 0 5.900000

E.y=X%\*%mle2hr.opt$par[1:4]

# Residuals

library(TSA)

resids=data2hr$weights-E.y

# Run test

runs(resids,0)$pvalue

[1] 2.8e-07

plot(density(resids))

# Goodness-of-fits (KS on residuals)

ks.test(resids,"pnorm",alternative="two.sided")

One-sample Kolmogorov-Smirnov test

data: resids

D = 0.11368, p-value = 0.04038

alternative hypothesis: two-sided

# residuals vs fitted

pos=(resids)>0

col=rep("blue",length(resids))

col[pos]="red"

plot(E.y,resids,col=col,pch=19)

abline(0,0,lty=2)

# QQ plot

qqnorm(resids)

qqline(resids)

# glm for model 4

fit4=glm(weights~first+second+hours,family=gaussian,data=data2hr)

summary(fit4)

anova(fit4,test="F")

1-pchisq(deviance(fit4),df.residual(fit4))

plot(residuals(fit4,type="response")~predict(fit4,type="response"),xlab=expression(hat(mu)),ylab="Raw residuals")

plot(residuals(fit4,type="deviance")~predict(fit4,type="response"),xlab=expression(hat(mu)),ylab="Deviance residuals")

plot(residuals(fit4,type="pearson")~predict(fit4,type="response"),xlab=expression(hat(mu)),ylab="Pearson residuals")

# lm for model 4

fit4lm=lm(weights~first+second+hours,data=data1hr)

summary(fit4lm)

ols\_plot\_resid\_qq(fit4lm)

ols\_test\_normality(fit4lm)

ols\_test\_correlation(fit4lm)

ols\_plot\_resid\_fit(fit4lm)

ols\_plot\_resid\_hist(fit4lm)

plot(fit4lm)