1. Use times to failure for high speed turbine engine bearings made from two different compounds.

(a) Make normal plots for the lifetimes and log-lifetimes.

i. We do not expect “constant variance/normal distribution” ordinary statistical methods to be reliable in the analysis of these data because it seems compound 2 cannot be described with a normal distribution.

![Normal Q-Q Plot for compound 1](image1.png)

![Normal Q-Q Plot for compound 2](image2.png)

ii. It is still not appropriate to use “constant variance/normal distribution” ordinary statistical methods in the analysis of the log lifetimes.

![Normal Q-Q Plot for log(compound 1)](image3.png)

![Normal Q-Q Plot for log(compound 2)](image4.png)

iii. > median(compound2)
   [1] 4.68
   > B <- 10000
   > comp2boot.non <- bootstrap(compound2,B,"median")
   > round(sqrt(var(comp2boot.non)),3)
   [1] 0.831  # standard error for the sample median

(b) > kl <- floor((B+1)*.025)
   > ku <- B+1-kl
   > sortcomp2boot.non <- sort(comp2boot.non)
   > c(sortcomp2boot.non[kl],sortcomp2boot.non[ku])
   [1] 4.395 7.575  # 95% percentile bootstrap confidence interval for the median of F

(c) The ML estimates of the shape and scale parameters of a Weibull distribution are 2.32 and 6.86, respectively.

> fit2 <- fitdistr(compound2,"weibull")
> fit2
   shape    scale
   2.3200021 6.8594860
   (0.5243508) (0.9958011)
A parametric bootstrap standard error for the sample median is 1.073 millions of cycles and a parametric 95% (unadjusted) percentile bootstrap confidence interval for the median of F is (3.897, 8.063). The parametric standard error is larger than the non-parametric standard error. Therefore, the parametric confidence interval is wider than the non-parametric confidence interval.

```
> comp2boot.Wei <- Wboot(10,B,"median",fit2$estimate[1],fit2$estimate[2])
> round(sqrt(var(comp2boot.Wei)),3)
[1] 1.073
> sortcomp2boot.Wei <- sort(comp2boot.Wei)
> round(c(sortcomp2boot.Wei[kl],sortcomp2boot.Wei[ku]),3)
[1] 3.897 8.063
```

(d) \(\frac{1}{2/(\theta)\sqrt{n}}\) = 1.169 is similar to the parametric bootstrap standard error.

```
> 1/(2*dweibull(median(compound2),fit2$estimate[1],fit2$estimate[2])*sqrt(10))
[1] 1.169048
```

(e) A 95% percentile confidence interval for the difference in underlying median lifetimes is (−10.54, −0.66). This difference is clearly non-zero.

```
> comp1boot.non <- bootstrap(compound1,B,"median")
> diff.non <- comp2boot.non-comp1boot.non
> c(sort(diff.non)[kl],sort(diff.non)[ku])
[1] -10.54 -0.66
```

2. See Prof. Vardeman’s solution posted on the 2003 Stat 511 Web page.

1. Use data set of pre-Challenger space shuttle flights.

(a) > summary(shuttle.out)
Call:
glm(formula = indicate ~ temp, family = binomial)

Coefficients:

                   Estimate Std. Error z value Pr(>|z|)
(Intercept)      15.0429     7.3719   2.041  0.0413 *
temp             -0.2322     0.1081  -2.147  0.0318 *

---
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 28.267 on 22 degrees of freedom
Residual deviance: 20.315 on 21 degrees of freedom
AIC: 24.315

Number of Fisher Scoring iterations: 4

There is evidence that the coefficient of the temperature covariate is non zero (p-value=0.03). The test $H_0: \beta_1 < 0$ has a p-value of 0.015. I would say that this data set suggests that their claim was not correct.

(b) shuttle.fits <- predict.glm(shuttle.out,type="response",se.fit=TRUE)

ind <- order(temp)
plot(temp[ind],shuttle.fits$fit[ind],type="b",ylim=c(0,1),
    xlab="Temperature",ylab="Estimated means",main="Relationship between t and p")
lines(temp[ind],shuttle.fits$fit[ind]-2*shuttle.fits$se.fit[ind],lty=2)
lines(temp[ind],shuttle.fits$fit[ind]+2*shuttle.fits$se.fit[ind],lty=2)

The temperature 31°F is outside the range of temperature values used to fit the model. However, assuming that the relationship of temperature and O-ring incidents remains the same at lower temperature values, the model suggests that there is a very high probability of having an O-ring incident on a launch at 31°F.

> predict.glm(shuttle.out,data.frame(temp=31),se.fit=TRUE,type="response")

$fit
[1] 0.9996088

$se.fit
[1] 0.001578722
Use data set gathered in a project aimed at reducing jams on a large collating machine.

(a) It appears that there are statistically detectable Air Pressure and Bar Tightness effects in these data since the coefficients for the levels of these two variables are significant. If one wants small number of jams, one wants level 2 of Air Pressure and level 2 of Bar Tightness.

```r
> summary(collator.out)
```

```
Call:
glm(formula = y ~ AA + BB, family = poisson, offset = log(k))

Deviance Residuals:
1 2 3 4 5 6
-0.5040 0.1693 0.3442 0.7453 -0.3004 -0.5532

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             -3.2159    0.1068  -30.099  < 2e-16 ***
AA1                     0.2420    0.1313   1.844  0.06520 .
AA2                     -0.4856    0.1472  -3.299  0.00097 ***
BB1                     0.6781    0.1045   6.491  8.54e-11 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 59.201 on 5 degrees of freedom
Residual deviance: 1.353 on 2 degrees of freedom
AIC: 37.188

Number of Fisher Scoring iterations: 3
```

(b) `mu` <- `collator.out$coefficients[1]`
`alpha3` <- -(`alpha1`+`alpha2`)
`beta1` <- `collator.out$coefficients[4]` ; `beta2` <- `-beta1`

```r
> exp(c(mu+alpha1+beta1, mu+alpha2+beta1, mu+alpha3+beta1, 
>    mu+alpha1+beta2, mu+alpha2+beta2, mu+alpha3+beta2))
```

```
0.10069307 0.04863886 0.10084992 0.02593996 0.01253006 0.02598037
```

(c) `collator.fits$fit` = `k*(answer in 4(b))`.

```r
> collator.fits$fit
```

(d) `1collator.fits$fit`

```
1 2 3 4 5 6
3.391297 3.007353 3.435978 2.509237 1.911944 2.560186
```

See Prof. Vardeman’s solutions posted on the 2003 Stat 511 Web page.

---

7.
From the smoothing regressions below, apparently changing of temperature can be seen from the plots. The smooth help us determine this changing over time.

We first apply Kernel method here, use Cross validation to select bandwidth. The following plot is the CV value against bandwidth. We see here at bandwidth=4.15, the CV value attain its minimum. Therefore, we use this bandwidth in our kernel regression. We also try other bandwidth to do the nonparametric regression. For bandwidth=2 and bandwidth=10.

The following plot is the CV against bandwidth and kernel regression using bandwidth=4.15

library(faraway)
library(sm)
attach(aatemp)
h <- hcv(year, temp, hstart=2, hend=10, display="lines", ngrid=50)
sm.regression(aatemp$year, aatemp$temp, h=h, xlab="Year", ylab="Temperature ")
plot(aatemp$year, aatemp$temp, main="bandwidth=2",
    xlab="Year", ylab="Temperature", pch="." )
lines(ksmooth(aatemp$year, aatemp$temp, "normal", 2))
plot(aatemp$year, aatemp$temp, main="bandwidth=10",
    xlab="Year", ylab="Temperature", pch="." )
lines(ksmooth(aatemp$year, aatemp$temp, "normal", 10))
Secondly, we apply *Splines method*:

```r
plot(temp~year,aatemp,pch=".")
lines(smooth.spline(aatemp$year,aatemp$temp))

plot(temp~year,aatemp,pch=".",main="Different number of knots")
lines(smooth.spline(aatemp$year,aatemp$temp,nknots=10))
lines(smooth.spline(aatemp$year,aatemp$temp,nknots=6),lty=4,col=2)
lines(smooth.spline(aatemp$year,aatemp$temp,nknots=16),lty=5,col=4)
legend("topleft",c("Knots=10","Knots=6","Knots=16"),lty=c(1,4,5),col=c(1,2,4))

plot(temp~year,aatemp,pch=".",main="Different Lambda")
lines(smooth.spline(aatemp$year,aatemp$temp,spar=0.5))
lines(smooth.spline(aatemp$year,aatemp$temp,spar=0.3),lty=4,col=2)
lines(smooth.spline(aatemp$year,aatemp$temp,spar=0.9),lty=5,col=4)
legend("topleft",c("Lambda=0.5","Lambda=0.3","Lambda=0.9"),lty=c(1,4,5),col=c(1,2,4))
```
Thirdly, we apply LOWESS method,
plot(year, temp, main = "LOWESS with different spans")
lines(lowess(year, temp, f=.1), lty=1, col = 1)
lines(lowess(year, temp, f=.2), lty=4, col = 2)
lines(lowess(year, temp, f=.5), lty=5, col = 4)
legend("topleft", c("Span=0.1", "Span=0.2", "Span=0.5"), lty = c(1,4,5),
col = c(1,2,4))