HW11

Shaojie Zhang (shaojiez)

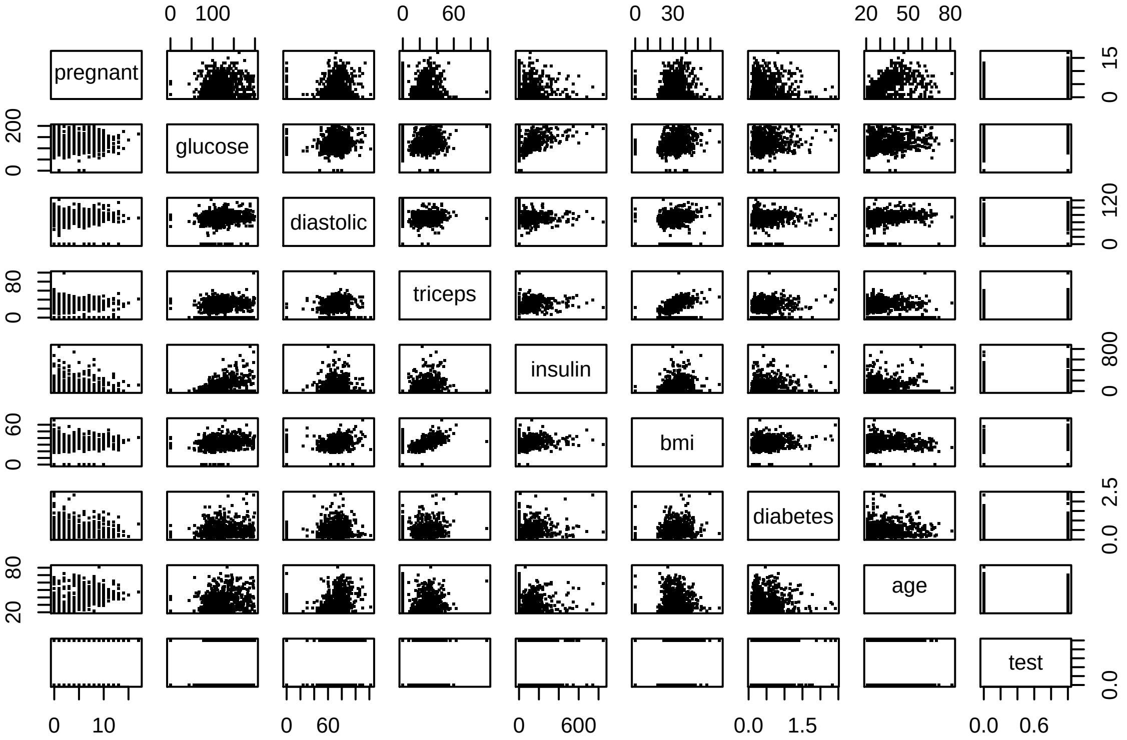
04/15/2019

Part a

```
pima=read.csv("pima.csv", header=T)

summary(pima)
```

```
##     pregnant         glucose        diastolic        triceps     
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00  
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00  
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00  
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54  
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00  
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00  
##     insulin           bmi           diabetes           age       
##  Min.   :  0.0   Min.   : 0.00   Min.   :0.0780   Min.   :21.00  
##  1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00  
##  Median : 30.5   Median :32.00   Median :0.3725   Median :29.00  
##  Mean   : 79.8   Mean   :31.99   Mean   :0.4719   Mean   :33.24  
##  3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00  
##  Max.   :846.0   Max.   :67.10   Max.   :2.4200   Max.   :81.00  
##       test      
##  Min.   :0.000  
##  1st Qu.:0.000  
##  Median :0.000  
##  Mean   :0.349  
##  3rd Qu.:1.000  
##  Max.   :1.000
```

```
pairs(pima, pch=".")
```



The 0 for some of those values probably means that there are missing data because it's impossible for example to have a 0 bicep.

```
pima$glucose[pima$glucose==0]=NA
pima$diastolic[pima$diastolic==0]=NA
pima$triceps[pima$triceps==0]=NA
pima$insulin[pima$insulin==0]=NA
pima$bmi[pima$bmi==0]=NA

pima.na=apply(pima, 1, function(x){any(is.na(x))})
npima=pima[!pima.na,]
```

Part b

```
Model1=glm(test~pregnant+glucose+diastolic+triceps+insulin+bmi+diabetes+age, family=binomial,data=npima
summary(Model1)
```

```
##
## Call:
## glm(formula = test ~ pregnant + glucose + diastolic + triceps +
##     insulin + bmi + diabetes + age, family = binomial, data = npima)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7823  -0.6603  -0.3642   0.6409   2.5612
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.004e+01  1.218e+00  -8.246  < 2e-16 ***
## pregnant     8.216e-02  5.543e-02   1.482  0.13825
## glucose      3.827e-02  5.768e-03   6.635 3.24e-11 ***
## diastolic   -1.420e-03  1.183e-02  -0.120  0.90446
## triceps      1.122e-02  1.708e-02   0.657  0.51128
## insulin     -8.253e-04  1.306e-03  -0.632  0.52757
## bmi          7.054e-02  2.734e-02   2.580  0.00989 **
## diabetes     1.141e+00  4.274e-01   2.669  0.00760 **
## age          3.395e-02  1.838e-02   1.847  0.06474 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 344.02  on 383  degrees of freedom
## AIC: 362.02
##
## Number of Fisher Scoring iterations: 5
```

From the summary, we see that glucose, bmi and diabetes are the ones that contribute.

Part c

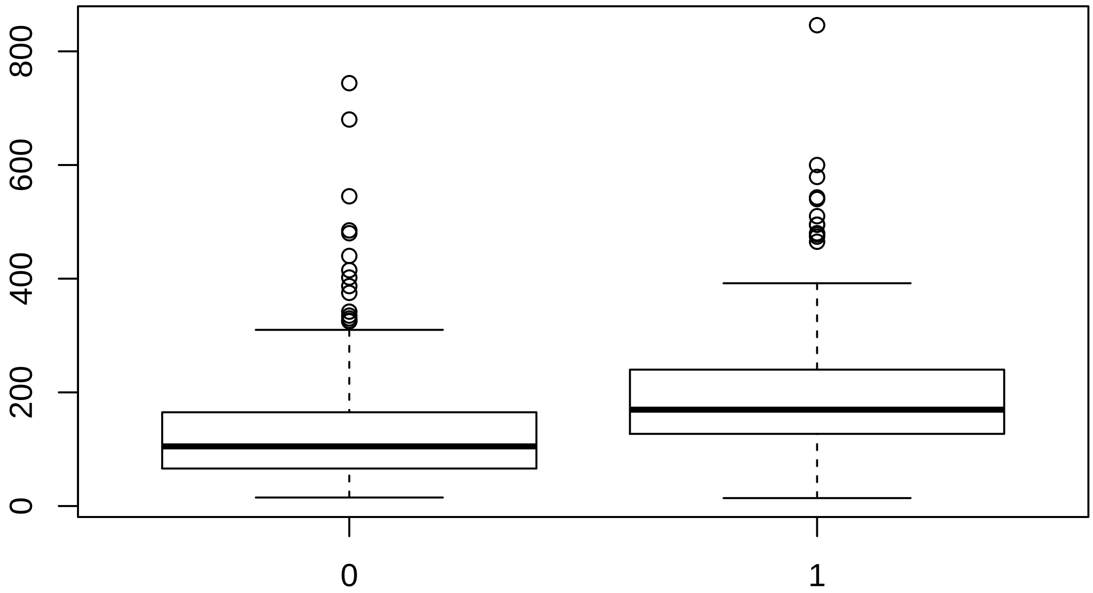
```r
Model2=glm(test~1, family=binomial, data=npima)

anova(Model2, Model1, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: test ~ 1
## Model 2: test ~ pregnant + glucose + diastolic + triceps + insulin + bmi +
##     diabetes + age
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       391     498.10
## 2       383     344.02  8   154.08 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note here that for binomial model we should use Chisquare test. From the test result we see that model1 is a great improvement.

Part d

```r
with(npima, plot(factor(test), insulin))
```



From the boxplot, we see that within the ones that have diabetes, insulin is higher. The coeff in model1 for insulin is negative but this does not contradict with each other because the potential correlation between insulin and the other variables in the model.

Part e

```r
Model3=step(Model1, direction="backward", trace=0)
anova(Model3, Model1, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: test ~ pregnant + glucose + bmi + diabetes + age
## Model 2: test ~ pregnant + glucose + diastolic + triceps + insulin + bmi +
##     diabetes + age
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       386     344.89
## 2       383     344.02  3   0.8639   0.8341
```

From the deviance test, we see that there is barely any improvement. So either model1 and model3 could be the best among these three models.

Part f

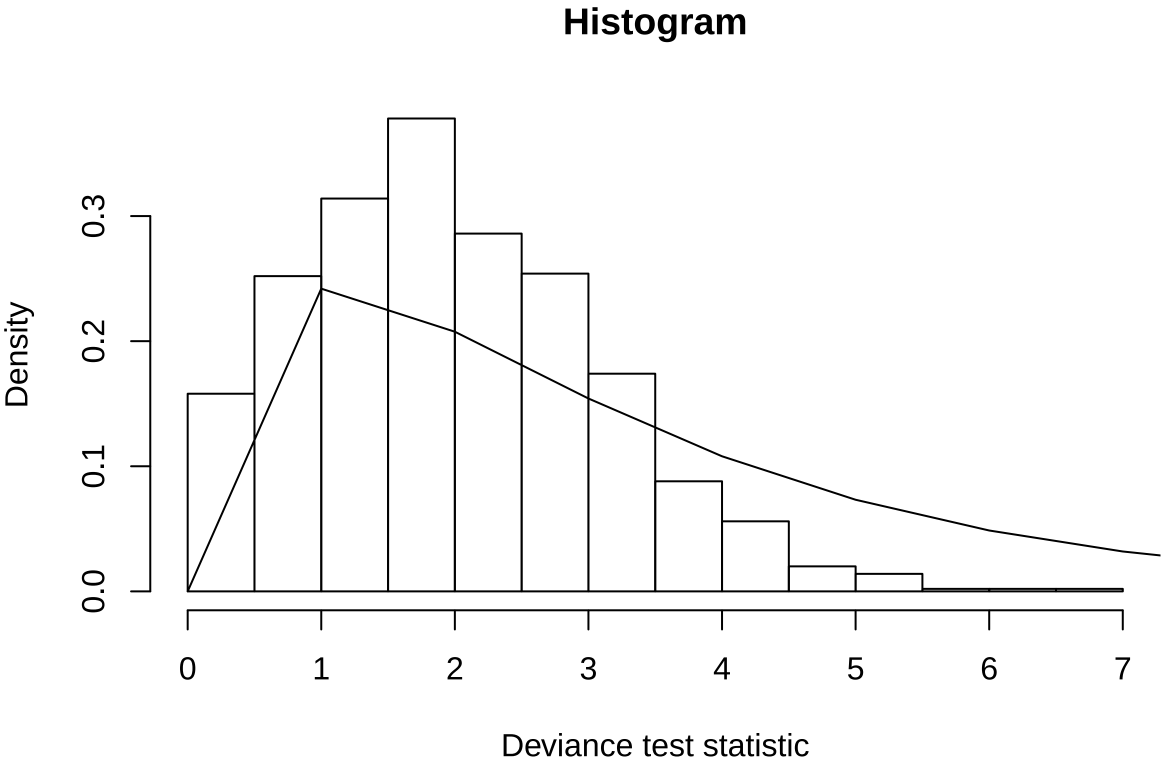
```r
samp=npima
tresult=NULL

for(b in 1:1000){
  samp$test=rbinom(nrow(samp), 1, fitted.values(Model3))
  m1=glm(test~pregnant+glucose+diastolic+triceps+insulin+bmi+diabetes+age, family=binomial, data=samp)
  m3=step(m1, direction="backward", trace=0)
  tresult[b]=anova(m3, m1, test="Chisq")$Deviance[2]
}
```

Now compare:

```r
test1 = anova(Model3, Model1, test="Chisq")
mean(tresult>=test1$Deviance[2])
```

```
## [1] 0.829
```

```r
hist(tresult, freq = F, xlab="Deviance test statistic", main="Histogram")
lines(c(0:65), dchisq(c(0:65), 3))
```



Notice that the quantile is at 0.65 for model3, and degrees of freedom is 3. So from the comparison we see that the trends are pretty much the same.

Part g

```r
woman=data.frame(pregnant=3, glucose=103, diastolic=70, triceps=29.2, insulin=160, bmi=32.4, diabetes=0
wpred=predict(Model3, woman, "response", se.fit=T)
wpred$fit
```

```
##         1
## 0.1593196
```

So the predicted test result is 0.1593196. And the confidence interval:

```r
c(wpred$fit + wpred$se.fit*qnorm(0.05), wpred$fit + wpred$se.fit*qnorm(0.95))
```

```
##         1         1
## 0.1159205 0.2027186
```

Part h

Part i

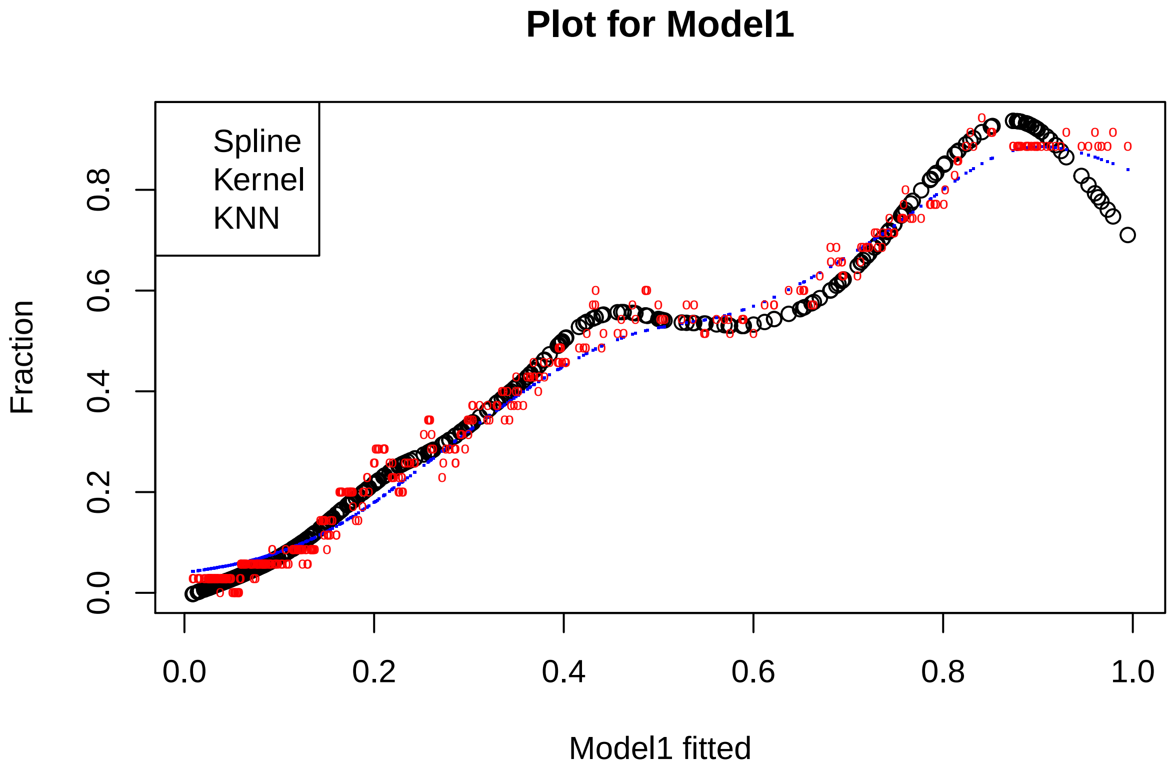
```r
library(np)
```

```
## Warning: package 'np' was built under R version 3.4.4
## Nonparametric Kernel Methods for Mixed Datatypes (version 0.60-9)
## [vignette("np_faq",package="np") provides answers to frequently asked questions]
## [vignette("np",package="np") an overview]
## [vignette("entropy_np",package="np") an overview of entropy-based methods]
```

```
library(FNN)
```

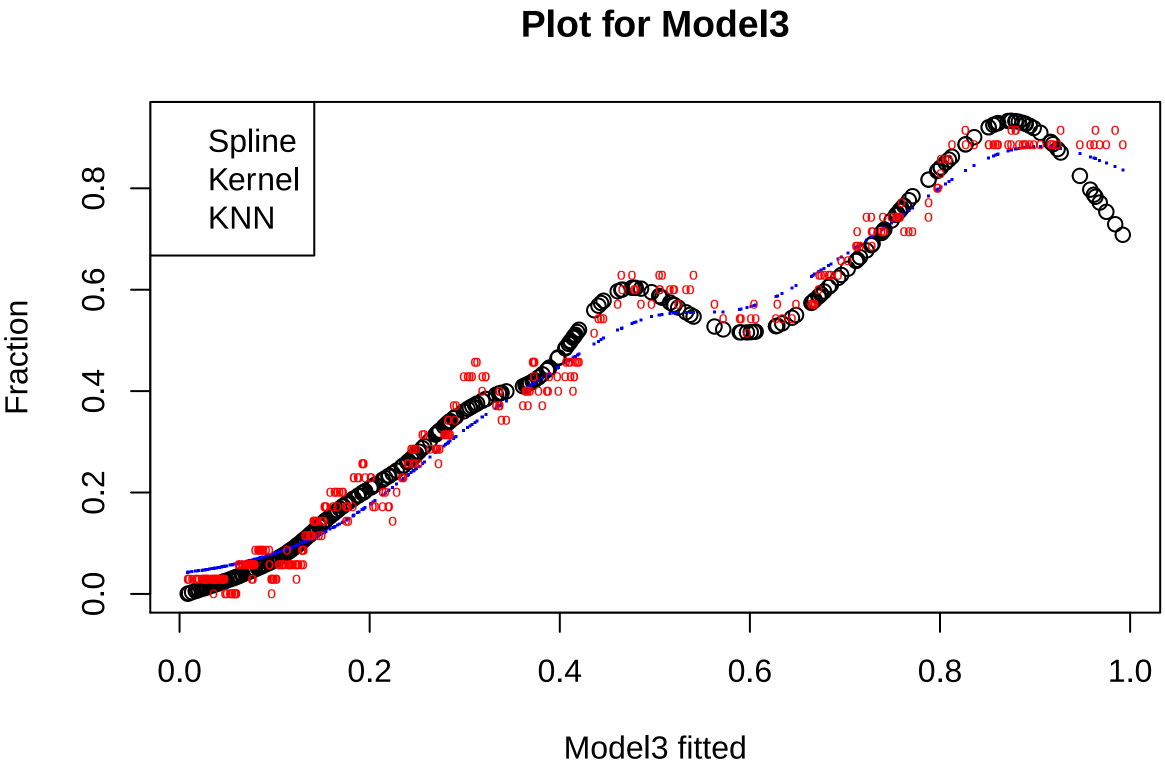
```
## Warning: package 'FNN' was built under R version 3.4.4
```

```
# model1
spline1=smooth.spline(npima$test~fitted.values(Model1), df=10)
kernel1=npreg(npima$test~fitted.values(Model1), bws=0.075)
knn1=knn.reg(fitted.values(Model1), y=npima$test, k=35)

plot(fitted.values(Model1), fitted.values(spline1), xlab="Model1 fitted", main="Plot for Model1", ylab=
points(fitted.values(Model1), fitted.values(kernel1), pch=".", col="blue")
points(fitted.values(Model1), knn1$pred, pch="o", cex = 0.5, col="red")
legend("topleft", col=c("black","blue","red"), legend=c("Spline","Kernel","KNN"))
```



And here is model3:

```
spline3=smooth.spline(npima$test~fitted.values(Model3), df=10)
kernel3=npreg(npima$test~fitted.values(Model3), bws=0.075)
knn3=knn.reg(fitted.values(Model3), y=npima$test, k=35)

plot(fitted.values(Model3), fitted.values(spline3), xlab="Model3 fitted", main="Plot for Model3", ylab=
points(fitted.values(Model3), fitted.values(kernel3), pch=".", col="blue")
points(fitted.values(Model3), knn3$pred, pch="o", cex = 0.5, col="red")
legend("topleft", col=c("black","blue","red"), legend=c("Spline","Kernel","KNN"))
```



From the plots, we see that the two plots are super similar.