Department of Statistics

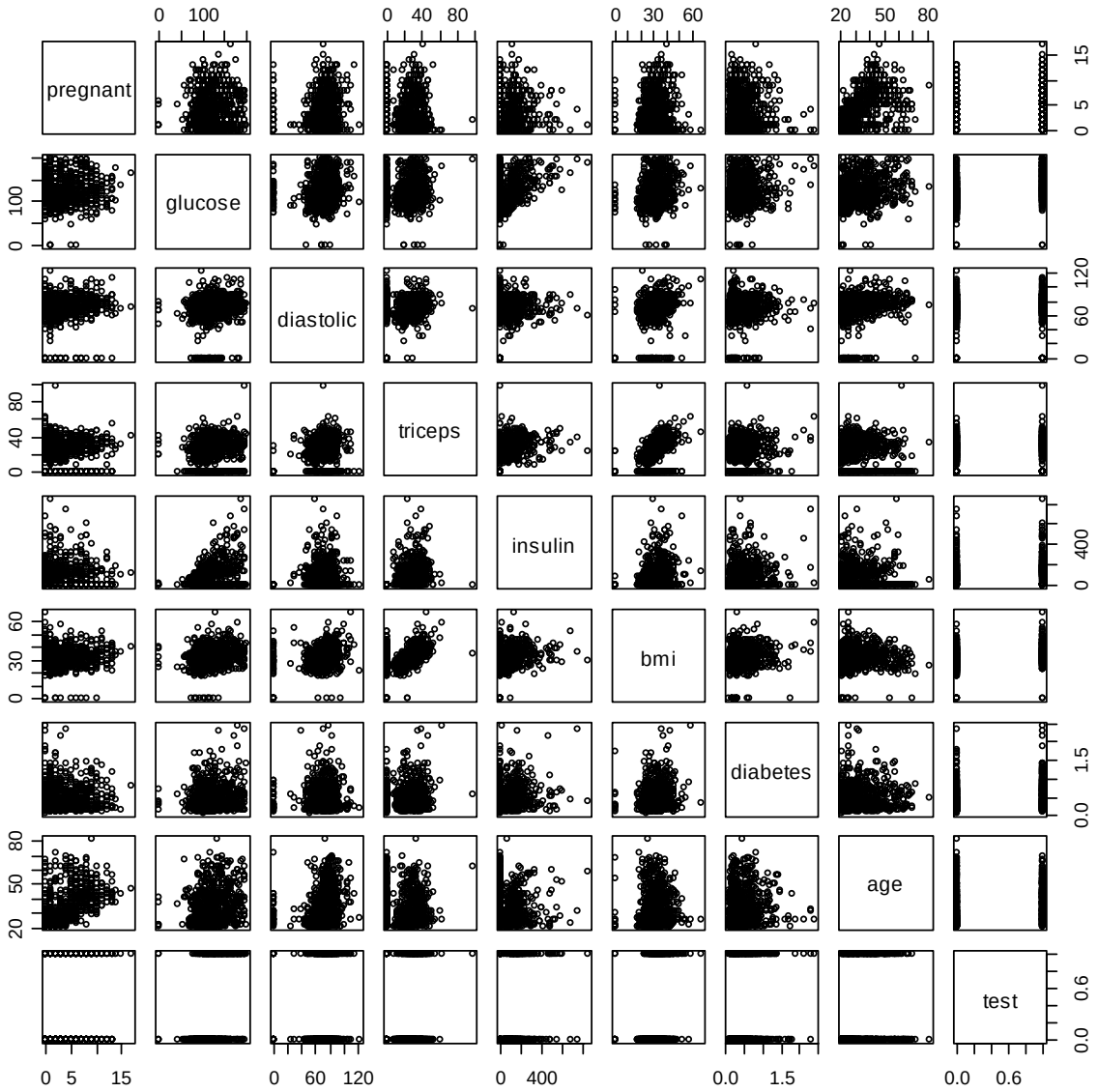
COURSE STATS 330

Model answer to Assignment 4, 2006

Question 1

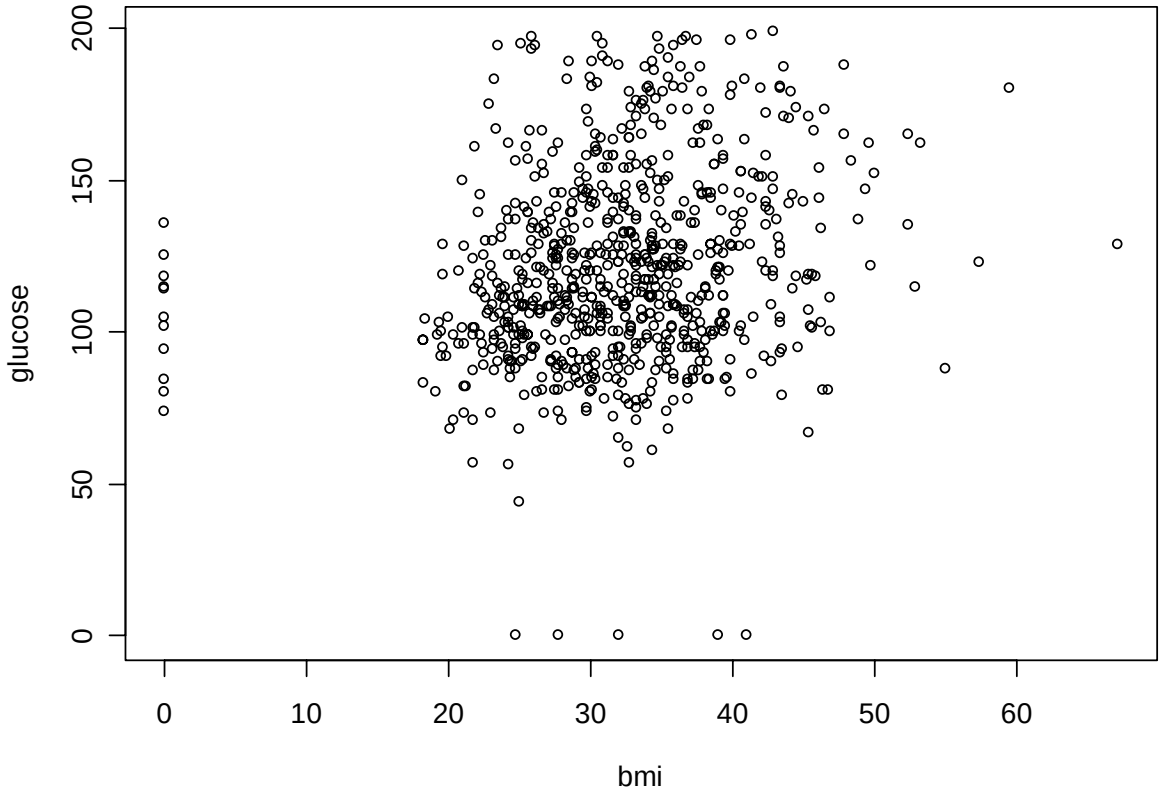
(a) Read the data into R and make a pairs plot. These data have an unusual feature, which you should diagnose and correct before proceeding further.

```
pima.df = read.table(file.choose(), header=T)
pairs(pima.df)
```



There seem to be a lot of zeros in the data. For example, if we plot **glucose** versus **bmi**, we get

```
plot(glucose~bmi, data=pima.df)
```



It seems that zero is being used as a missing value code - it may be possible to have zero glucose, but its not possible to have zero BMI!!!

The affected variables are **glucose, diastolic, triceps, insulin, bmi**, and **age**. The number of zeros in each variable can be calculated by e.g.

```
sum(pima.df$glucose==0)
```

continuing, we get

```
> sum(pima.df$glucose==0)
[1] 5
> sum(pima.df$diastolic==0)
[1] 35
> sum(pima.df$triceps==0)
```

```
[1] 227
> sum(pima.df$insulin==0)
[1] 374
> sum(pima.df$bmi==0)
[1] 11
> sum(pima.df$diabetes==0)
[1] 0
> sum(pima.df$age==0)
[1] 0
```

Are the zeros genuine or missing value codes? Ideally we should ask the people who coded the data. Since this is not possible, we can look at the distributions of the data for these variables, as we did for **`glucose`** and **`bmi`** above.

For these variables, zero looks like a missing value code, as there is a gap between the real data and the zeros.

On this basis, **`glucose, diastolic, triceps`**, and **`bmi`** are missing values. In any event, we know that the true values of **`diastolic`** and **`bmi`** can't be zero. (If you have zero blood pressure you are dead, and if you have zero weight you don't exist!) There are too many zeros to be mistakes. We will assume they are missing value codes.
[3 marks for this conclusion.]

We will change them to NA's so that R will delete the correct cases when fitting models. If we want to compare models rather than just fit them we have to delete cases manually as we did in assignment 2. The affected variables are

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

[3 marks identifying these variables, and 4 marks for changing them to NA's .]

(b) *Fit a logistic regression model to the data, using test as the response and the other variables as explanatory variables. Do any of the explanatory variables need transforming? Are there any influential points? Make any adjustments to your model you feel necessary.*

First, let's fit a model to all the variables.

```
> pima.glm = glm(test~., family=binomial, data=pima.df)
> summary(pima.glm)

Call:
glm(formula = test ~ ., family = binomial, data = pima.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8627  -0.6639  -0.3672   0.6347   2.4942
```

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.677562   1.005400  -9.626  < 2e-16 ***
pregnant     0.121235   0.043926   2.760 0.005780 **
glucose      0.037439   0.004765   7.857 3.92e-15 ***
diastolic   -0.009316   0.010446  -0.892 0.372494
triceps      0.006341   0.014853   0.427 0.669426
insulin     -0.001053   0.001007  -1.046 0.295651
bmi          0.085992   0.023661   3.634 0.000279 ***
diabetes     1.335764   0.365771   3.652 0.000260 ***
age          0.026430   0.013962   1.893 0.058371 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 676.79  on 531  degrees of freedom
Residual deviance: 465.23  on 523  degrees of freedom
AIC: 483.23

Number of Fisher Scoring iterations: 5
```

Note that there are now only 532 observations used to fit the regression. We have assumed that the missing observations are typical of the rest (missing completely at random). It seems that the variables diastolic , insulin and insulin are not very important. *[2 marks for the initial fit]*

Next, we want to select the variables to go into our model. Unfortunately the step function will object to the NA's. If we just delete all the cases with missing data, we are wasting information. What else could we do? One possibility is to check if we can dispense with triceps and insulin, which account for most of the missing values. If this is the case (and it is) we could then eliminate the cases having missing values on the other variables, and perform a stepwise regression on the remainder.

A neater alternative is to do backward elimination by hand. Since we are only fitting a single model at each stage, R will handle the missing values automatically and use all the appropriate data at each stage.

The results are

1. for the full model, the least significant variable is triceps. Eliminate this.
2. For the model **`test~.-triceps,`** the least significant variable is insulin. Eliminate this.
3. For the model **`test~.-triceps - insulin,`** all the variables are significant at the 10% level.

The output for this model is

```
Call:
glm(formula = test ~ . - triceps - insulin, family = binomial,
    data = pima.df)
```

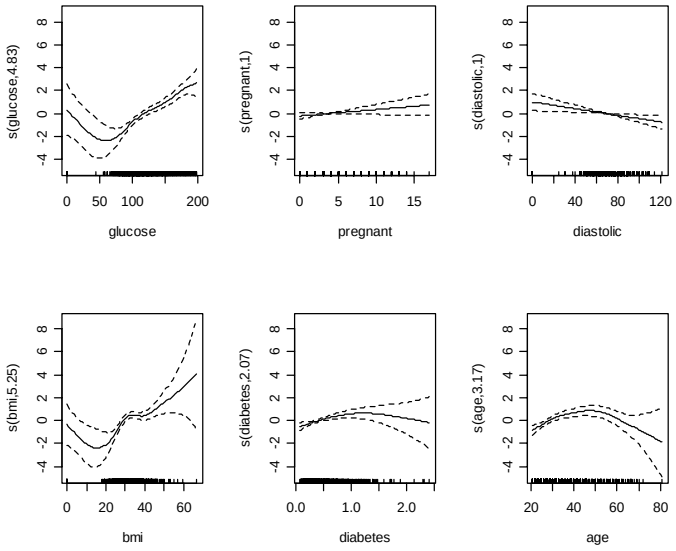
```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7380  -0.7313  -0.4123   0.7276   2.8984

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -8.239812   0.701970 -11.738  < 2e-16 ***
pregnant     0.124919   0.031972   3.907 9.34e-05 ***
glucose      0.033492   0.003440   9.736  < 2e-16 ***
diastolic   -0.013487   0.005114  -2.637  0.00836 **
bmi          0.087676   0.014268   6.145 7.99e-10 ***
diabetes     0.896150   0.294862   3.039  0.00237 **
age          0.016325   0.009237   1.767  0.07719 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 993.48  on 767  degrees of freedom
Residual deviance: 725.46  on 761  degrees of freedom
AIC: 739.46

Number of Fisher Scoring iterations: 5
```

We will use this model in what follows. Note that to fit this model, we can use 768 observations.

[3 marks for variable selection. You lost marks if you didn't attempt to make the best use of all the data.]

Should we transform any of the independent variables? We can look at gam plots:

```
library(mcgv)
par(mfrow=c(2,3))
plot(gam(test~  s(glucose) +s(pregnant) + s(diastolic) + s(bmi) +
s(diabetes) +s(age)family=binomial, data=pima.df))
```

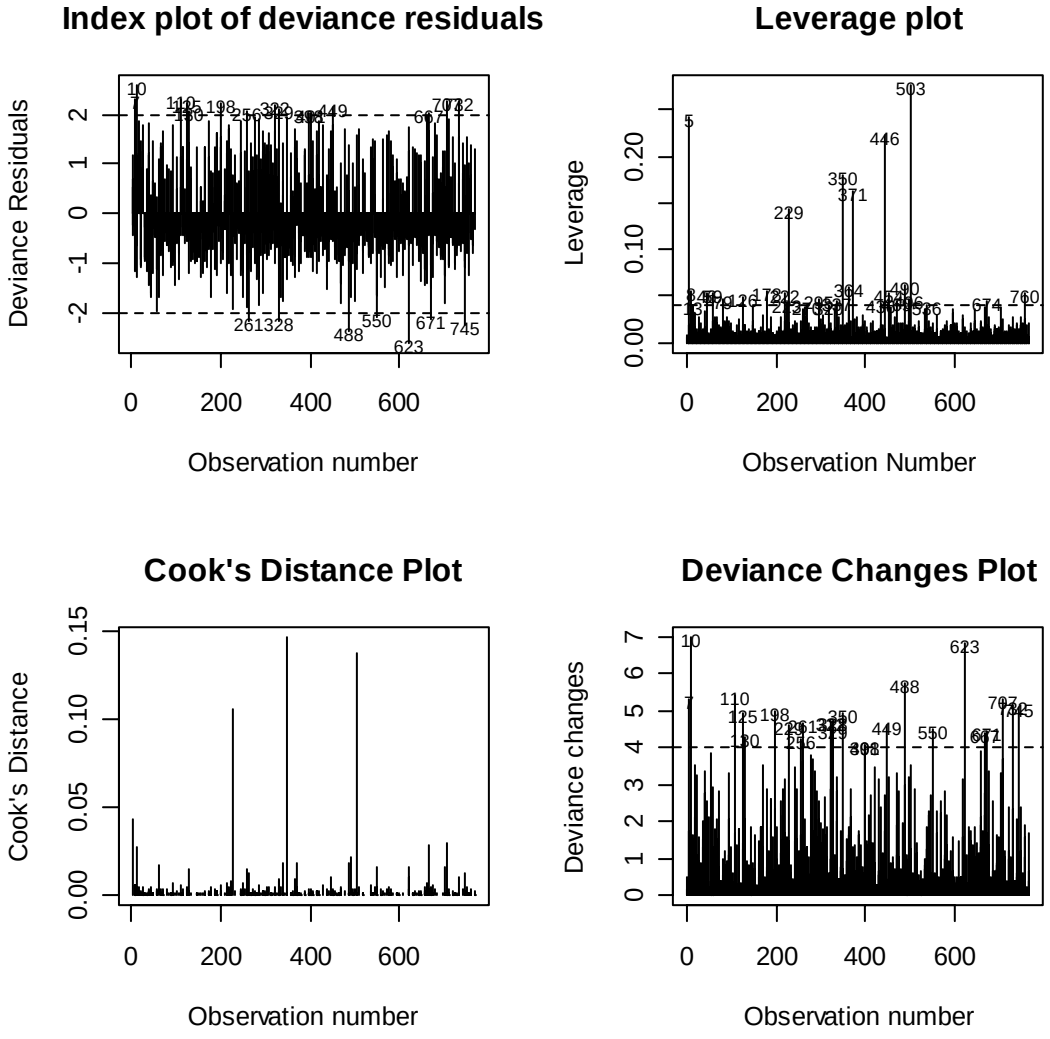


On the basis of these plots, it looks like **`glucose, bmi, diabetes`** and **`age`** might be transformed. Let's fit each as a quadratic. Checking the resulting summary, it looks like we could get rid of the second degree term in **`bmi`**. Our final model is

```
test ~ poly(glucose,2) + pregnant + diastolic + bmi + poly(diabetes, 2) +
poly(age, 2)
```

We could delete **`pregnant`** but I will keep it in (it was selected by stepwise).
[2 marks for transforming]

Lets draw some diagnostic plots.



There are quite a number of points indicated here – we will just check the top few, say points 10, 503, 623. The results of leaving out these points are shown overleaf.

	none	10	503	623	all
(Intercept)	**-11.21909**	**-11.35760**	**-12.70978**	**-11.29515**	**-12.79956**
glucose	**-0.00152**	**-0.00234**	**0.02361**	**-0.00381**	**0.01845**
I(glucose^2)	**0.00014**	**0.00014**	**0.00005**	**0.00015**	**0.00007**
pregnant	**0.05174**	**0.05246**	**0.05245**	**0.05227**	**0.05362**
diastolic	**-0.01498**	**-0.01618**	**-0.01488**	**-0.01471**	**-0.01582**
bmi	**0.07812**	**0.08516**	**0.07721**	**0.07878**	**0.08506**
diabetes	**2.68404**	**2.75463**	**2.60705**	**2.72745**	**2.72527**
I(diabetes^2)	**-1.21623**	**-1.25541**	**-1.16986**	**-1.19927**	**-1.19532**
age	**0.29935**	**0.29948**	**0.29481**	**0.30500**	**0.30081**

There are no dramatic changes so we will leave all the points in the analysis.
[3 marks for diagnostics and the sensitivity analysis.]

(c) *Develop a formula that will allow you to evaluate the probability that a randomly chosen Pima woman will have diabetes, given the values of the explanatory variables above. Make up an example (i.e. a set of values of the above variables) to illustrate the use of your formula, using suitable R code to evaluate your formula*

On the basis of the model above we can get a formula for the probability. The coefficients are

```
> coef(model3.glm)
> coef(model3.glm)
  (Intercept)       glucose  I(glucose^2)      pregnant     diastolic
-1.121909e+01 -1.518780e-03  1.412179e-04  5.174080e-02 -1.497653e-02
          bmi      diabetes I(diabetes^2)           age      I(age^2)
 7.812055e-02  2.684045e+00 -1.216225e+00  2.993490e-01 -3.365149e-03
```

The formula for the probability π is (rounding to 4 sig figs)

$\log(\pi/(1-\pi))$ = `-11.22 + 0.001519*glucose + 0.0001412*glucose`^{2} +
`0.05174*pregnant – 0.1498*diastolic + 0.07812*bmi + 2.864*diabetes`
`- 1.216*diabetes`^{2} + `0.2993*age - 0.003365*age`^{2}

To evaluate this for example for glucose = 92, pregnant = 6, diastolic = 70, bmi=19.1, diabetes=0.188, age =28, we type

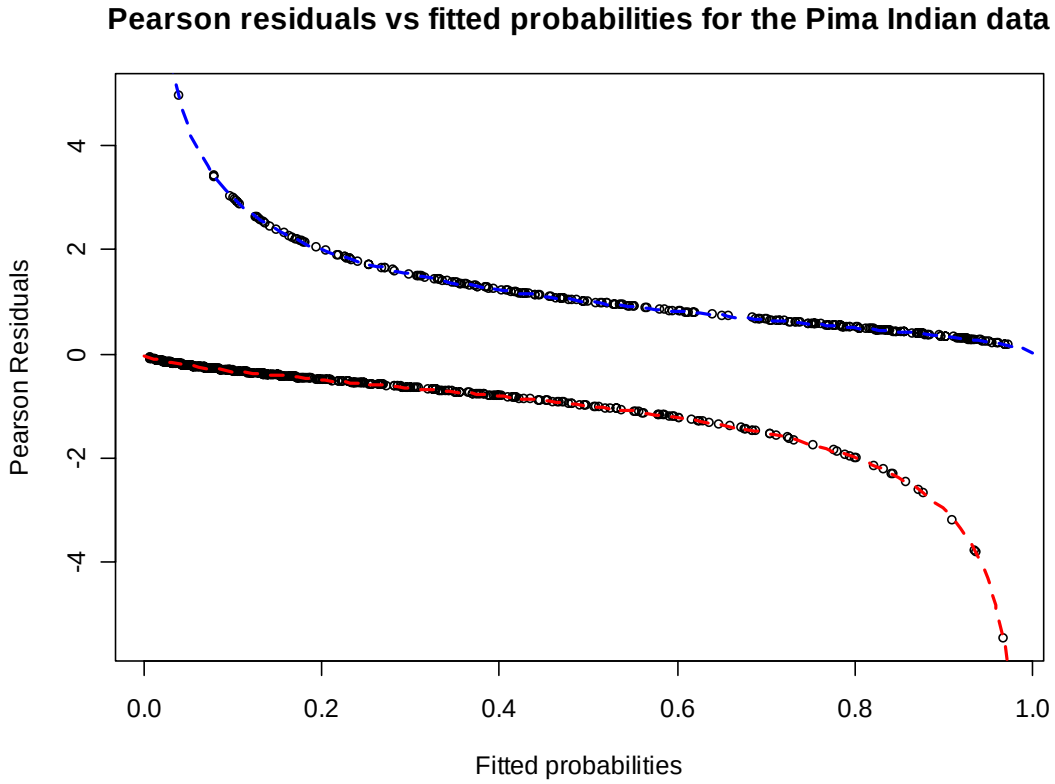
```
example.df=data.frame(glucose = 92, pregnant = 6, diastolic = 70, bmi=19.1,
diabetes=0.188, age =28)
predict(model3.glm, example.df, type="response")
[1] 0.03900596
```

About a 4% chance for this person.
[2 marks for the formula, 3 marks for the numeric example. Total for Q1: 25 marks].

Question 2

The plot of Pearson residuals versus fitted probabilities is calculated by

```
pearson.resids = residuals(model3.glm, type="pearson")
fitted.probs = predict(model3.glm, type="response")
plot(fitted.probs, pearson.resids, ylab="Pearson Residuals",
xlab="Fitted probabilities", main = "Pearson residuals vs
fitted probabilities for the Pima Indian data")
```



The Pearson residuals are

$$residual = \frac{r/n - \pi}{\sqrt{\frac{\pi(1-\pi)}{n}}} = \frac{r - n\pi}{\sqrt{n\pi(1-\pi)}}.$$

When n = 1 (i.e. when we have ungrouped data as we do here), the only possible values of r are 0 and 1. When r = 0, the residual is

$$\frac{0-\pi}{\sqrt{\pi(1-\pi)}} = -\sqrt{\frac{(1-\pi)}{\pi}} \quad .$$

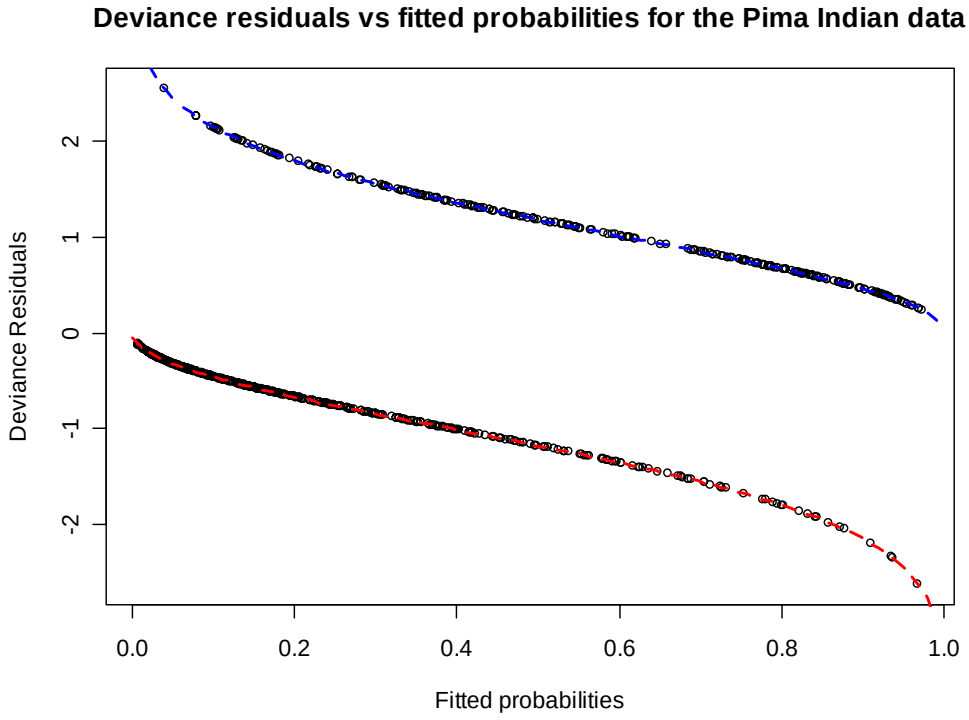
When r = 1, the residual is $\frac{1-\pi}{\sqrt{\pi(1-\pi)}} = -\sqrt{\frac{\pi}{(1-\pi)}}$. Thus the residuals cluster along the curves

$residual = -\sqrt{\frac{(1-\pi)}{\pi}}$ and $residual = \sqrt{\frac{\pi}{1-\pi}}$. These are shown on the plot as a dashed line.

[7 marks: 3 for the formula and 4 for the reason]

In a similar way, for ungrouped data the deviance residuals are $\sqrt{2\,|\log \pi\,|}$ when y = 1, and $-\sqrt{2\,|\log(1-\pi)\,|}$ when y = 0. These are shown on the plot below.

[7 marks: 3 for the formula and 4 for the reason]
[7 marks: 3 for the formula and 4 for the reason]



[8 marks: 5 for the formula and 3 for the reason. Total for Q2: 15 marks].