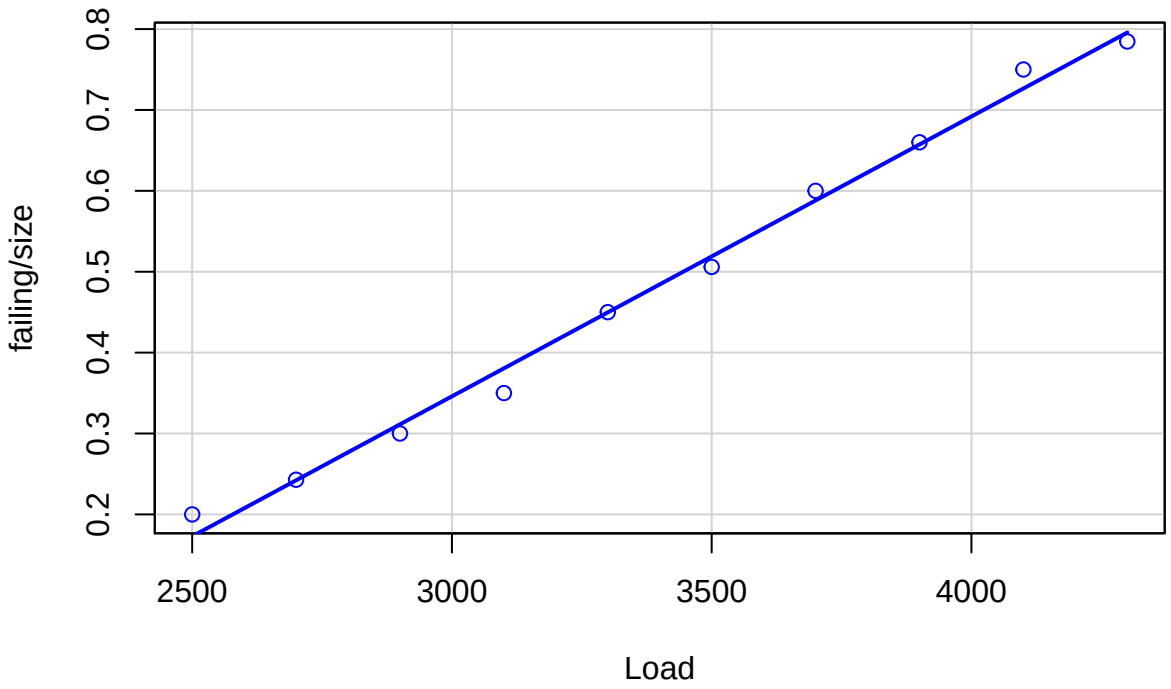
Production fails

A study of the compressive strength of an alloy fastener used in the construction of an aircraft was performed. Ten pressure loads, increasing in units of 200 psi from 2500 psi to 4300 psi, were used with different numbers of fasteners being tested at each of these loads. Plot the proportion of failures as a function of the loads.

```
library(car)
dd<-read.csv2("ex7.csv")
head(dd)
```

```
##   Load size failing
## 1 2500   50      10
## 2 2700   70      17
## 3 2900  100      30
## 4 3100   60      21
## 5 3300   40      18
## 6 3500   85      43
```

```
sp(failing/size~Load,dd,smooth=F,boxplot=F)
```



Define the GLM model that may be appropriate to fit these data and justify why it is appropriate. Assume that you already have your $\hat{\beta}$ vector. Deduce the formula to obtain the $\hat{p}_i$, that is the probability that the fasten fails at a given load pressure i. Fit the data with your model, and interpret the parameters of the model.

Since the sample size is defined and we want to analize the number of fails, the appropriate model would the Binomial. We use *cbind* to enter the number of "yes" (failed) and "no" (not failed). We use the canonical link, the link is defined inside the parentheses after the type of distribution. If leave the parentheses empty, it will use the canonical link by default. Also, if we want to use the canonical link, we can even leave out the parentheses:

```
m<-glm(cbind(failing,size-failing)~Load,family=binomial(link=logit),dd)
summary(m)
```

```
##
## Call:
## glm(formula = cbind(failing, size - failing) ~ Load, family = binomial(link = logit),
##     data = dd)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.29475  -0.11129   0.04162   0.08847   0.35016
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.3397115  0.5456932  -9.785   <2e-16 ***
## Load         0.0015484  0.0001575   9.829   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 112.83207  on 9  degrees of freedom
## Residual deviance:   0.37192  on 8  degrees of freedom
## AIC: 49.088
##
## Number of Fisher Scoring iterations: 3
```

```
# m$family
```

In LM the distributions are exact. In GLM the results are asymptotic (because the distribution is approximate). The dispersion parameter is $\phi = 1$. La deviància deguda al model és la diferència entre la Null Deviance i la Residual Deviance.

```
m$null.deviance-m$deviance
```

```
## [1] 112.4602
```

```
anova(m)
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: cbind(failing, size - failing)
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev
## NULL                     9    112.832
## Load  1   112.46         8      0.372
```

The linear predictor $\eta = X\beta$. With $logit(\mu) = \eta$ we can compute $p_i = \mu = logit^{-1}(\eta) = \frac{e^\eta}{1+e^\eta}$:

```
# Linear predictors:
(eta<-m$coefficients[1]+m$coefficients[2]*dd$Load)
```

```
##  [1] -1.4686274 -1.1589407 -0.8492540 -0.5395673 -0.2298805  0.0798062
##  [7]  0.3894929  0.6991797  1.0088664  1.3185531
```

```
# Predicted probabilities:
(p<-exp(eta)/(1+exp(eta)))
```

```
##  [1] 0.1871513 0.2388598 0.2995894 0.3682883 0.4427816 0.5199410 0.5961606
##  [8] 0.6680059 0.7327982 0.7889409
```

```
# The values are n*p:
(dd$size*p)
```

```
##  [1]  9.357566 16.720187 29.958938 22.097295 17.711265 44.194982 53.654456
##  [8] 33.400293 58.623859 51.281157
```

The same can be done with *predict*():

```
predict(m, ty="link")
```

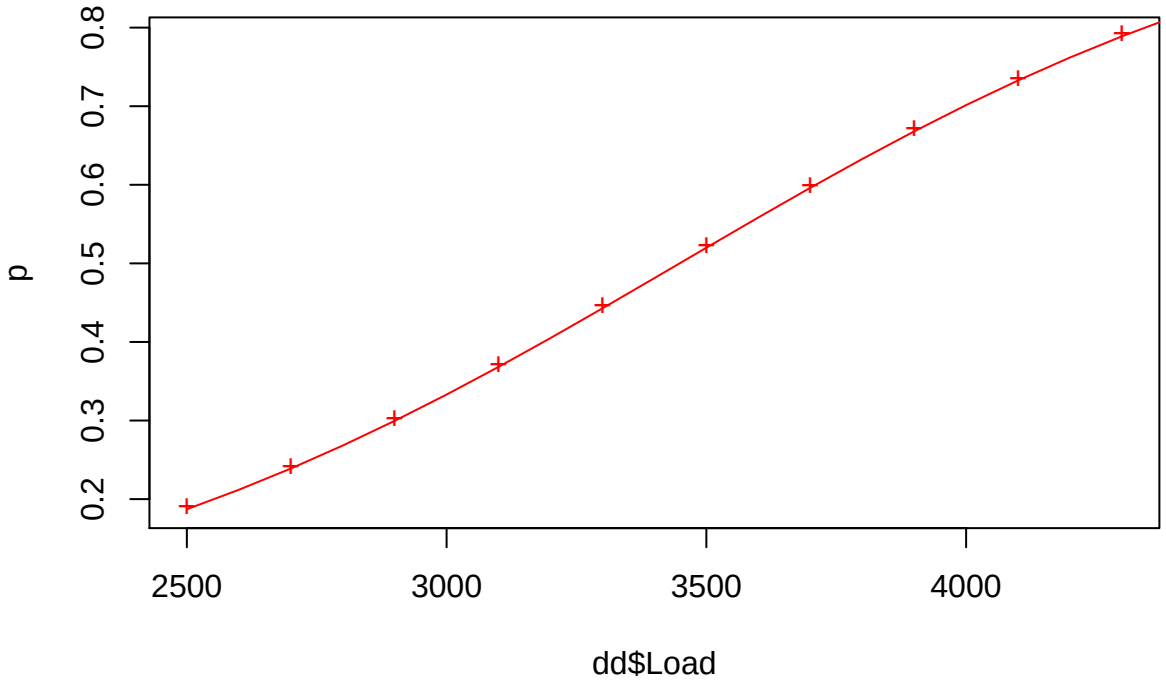
```
##          1          2          3          4          5          6
## -1.4686274 -1.1589407 -0.8492540 -0.5395673 -0.2298805  0.0798062
##          7          8          9         10
##  0.3894929  0.6991797  1.0088664  1.3185531
```

```
predict(m, ty="response")
```

```
##         1         2         3         4         5         6         7
## 0.1871513 0.2388598 0.2995894 0.3682883 0.4427816 0.5199410 0.5961606
##         8         9        10
## 0.6680059 0.7327982 0.7889409
```

The red points are the predictions that we have computed:

```
plot(dd$Load, p,col="red",pch="+")
Loads<-(25:45)*100
etas<-m$coefficients[1]+m$coefficients[2]*Loads
ps<-exp(etas)/(1+exp(etas))
lines(Loads,ps,col="red")
```



Let's calculate the Residuals: there are two types of residuals - the Pearson residuals and the Deviance residuals. The most intuitive ones are the Pearson's. We compute them as:

```
resid(m, ty="pearson")
```

```
##           1           2           3           4           5
##  0.232938865  0.078435844  0.008964054 -0.293693626  0.091909847
##           6           7           8           9          10
## -0.259434461  0.074232769 -0.120209277  0.347701061 -0.085461069
```

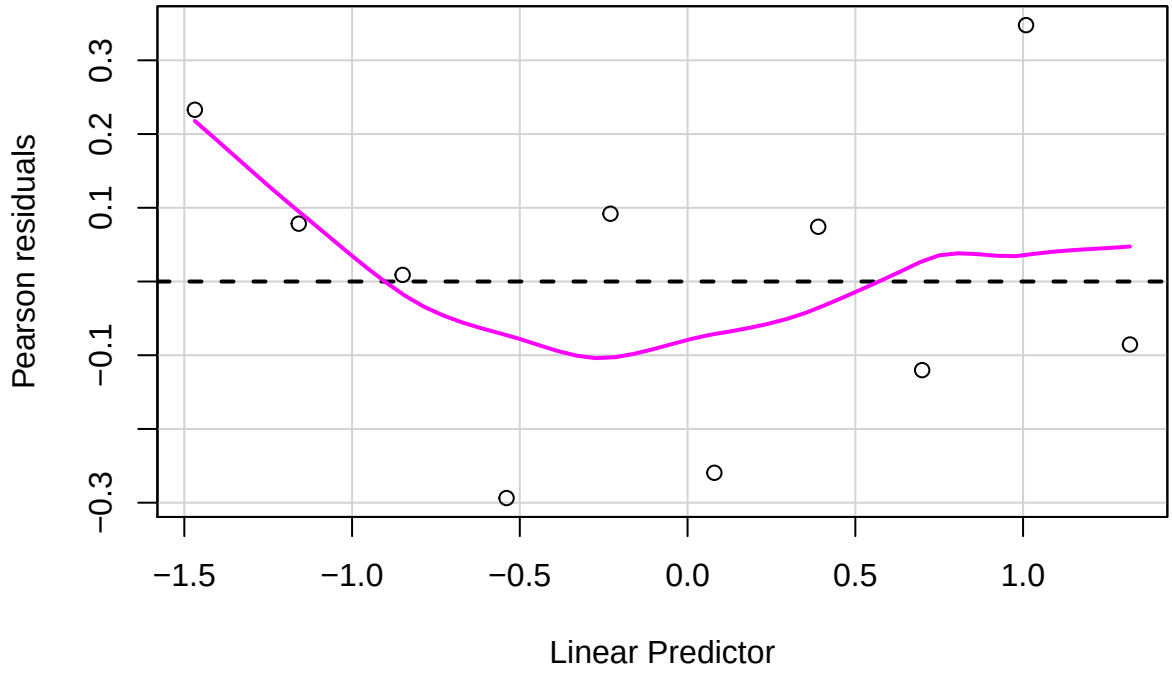
And the values are $\frac{y_i - \hat{\mu}_i}{\sqrt{V(\hat{\mu}_i)}}$:

```
n <- dd$size
(pearson.residuals<-(dd$failing-n*p)/sqrt(n*p*(1-p)))
```

```
##  [1]  0.232938865  0.078435844  0.008964054 -0.293693626  0.091909847
##  [6] -0.259434461  0.074232769 -0.120209277  0.347701061 -0.085461069
```

Per fer una gràfica de diagnòstic per veure si l'ajust és bo o no, podriem fer:

```
residualPlot(m, ty="pearson")
```



Perquè sigui bo, les variàncies han d'anar al voltant de zero i no hi ha d'haver patrons, la variància ha de ser constant.

Si haguèssim d'estimar el paràmetre de dispersió, l'estimariem a partir de l'estadístic de Pearson. També, si volem comprovar si sobredispersió o subdispersió, es faria mitjançant l'estadístic de Pearson. Càlcul de l'estadístic de Pearson (suma dels quadrats dels residuals de Pearson):

```
(pearson.statistic<-sum(pearson.residuals^2))
```

```
## [1] 0.3706631
```

Ha d'estar al voltant de 1. Podem fer el test de la chi quadrada: si acceptem $H_0 : X^2 = 1$ vol dir que no detectem ni sobredispersió ni subdispersió. Si rebutgem ka hipòtesi nul.la i acceptem $H_1 : X^2 \neq 1$ aleshores estem segurs que hi ha sobredispersió o subdispersió.

```r
pchisq(pearson.statistic*m$df.residual,m$df.residual) # cua inferior
```

```
## [1] 0.06348404
```

```r
pchisq(pearson.statistic*m$df.residual,m$df.residual,lower.tail=F) # cua superior (simètrica)
```

```
## [1] 0.936516
```

```r
# p-valor del 5%
(pvalor<-2*min(pchisq(pearson.statistic*m$df.residual,m$df.residual),
               pchisq(pearson.statistic*m$df.residual,m$df.residual,lower.tail=F)))
```

```
## [1] 0.1269681
```

EL p-valor ha sortit no significatiu, per tant acceptem hipòtesi nul.la. Si hi hagués alguna cosa, en tot cas seria subdispersió, ja que l'estadístic de Pearson ha sortit més petit que 1. La forma en general per calcular el p-valor seria dos vegades el mínim de la cua per l'esquerra i la cua per la dreta. O sinó també podem buscar l'interval de confiança amb chi quadrada de 2,5% dividint pels graus de llibertat:

```r
(IC95<-c(qchisq(0.025,m$df.residual)/m$df.residual,
         qchisq(0.975,m$df.residual)/m$df.residual))
```

```
## [1] 0.2724663 2.1918183
```

If a statistic is significantly different from 1 at the 0.05 level, then the 95% confidence interval will not contain 1. Since 1 is contained, it is not significantly different from 1 and we accept null hypothesi.