# Logistic Regession

F.A. Barrios

2020-12-03

```r
# All the needed libraries
library(tidyverse)
library(emmeans)
library(wesanderson)
library(rstatix)
library(HSAUR2)
library(car)
library(effects)

setwd("~/Dropbox/GitHub/Class2020")
wcgs <- read_csv("DataRegressBook/Chap2/wcgs.csv")
```

## Examples from the CAR book (Fox & Weisberg) [1]

### Review of the Structure of GLMs

The structure of a GLM is very similar to that of the linear model. In particular we have a response variable $y$ and $k$ predictors, and we are interested in understanding how the mean of $y$ varies as the values of the predictors change.

A GLM consists of three components

1. Random component, specifying the conditional or "error" distribution of the response variable, $y$, given the predictors from an *exponential family*. Both the binomial and Poisson distributions ae in the class of explonential families, and so problems with categorical or discrete responses can be studied with GLMs.
2. As in linear models, the $m$ predictors in a GLM are translated into a vector of $k+1$ regresor variables, $\mathbf{x} = (x_0, x_1, \ldots, x_k)$, possibly using contrast regressors for factors, polynomials, regression splines, transformations, and interactions. The response depends on the predictors only through a linear function of the regressors, called the *linear predictor*, $\eta(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$.
3. The connection between the conditional mean $E[y|\mathbf{x}]$ of the response and the predictor $\eta(\mathbf{x})$ in a linear model is direct,
$$E[y|\mathbf{x}] = \eta(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$$
and so the mean is equal to a linear combination of the regressors. This direct relation is not appropriate for all GLM because $\eta(\mathbf{x})$ can take any value, whereas the mean of a binary response variable must be in the interval (0,1). Therefore we introduce an invertible *link function* $g$ that translates from the scale of the mean response to the scale of the linear predictor. $\eta(\mathbf{x}) = E[y|\mathbf{x}]$ is standard in the GLM for the conditional mean of the response, therefore $g[\mu(\mathbf{x})] = \eta(\mathbf{x})$ Reversing this relationship produces the *inverse-link function*, $g^{-1}[\eta(\mathbf{x})] = \mu(\mathbf{x})$. The inverse of the link function is sometimes is sometimes called the *mean link funcion*

[1] All notes are taken form the "Companion to Applied Regression", 3rd Ed. Fox & Weisberg

Standard link functions and their inverses table: $\mu = E[y|\mathbf{x}]$ is the expected value of the response; $\eta = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$ is the linear predictor.

| **Link** | $\eta = \mathbf{g}(\mu)$ | $\mu = \mathbf{g^{-1}}(\eta)$ | **InverseLink** |
|---|---|---|---|
| *identity* | $\mu$ | $\eta$ | *identity* |
| *log* | $\log(\mu)$ | $e^{\eta}$ | *exponential* |
| *inverse* | $\mu^{-1}$ | $\eta^{-1}$ | *inverse* |
| *inversesquare* | $\mu^{-2}$ | $\eta^{-1/2}$ | *inversesquareroot* |
| *squareroot* | $\sqrt{\mu}$ | $\eta^2$ | *square* |
| *logit* | $\log \frac{\mu}{1-\mu}$ | $\frac{1}{1+e^{-\eta}}$ | *logistic* |
| *probit* | $\Phi(\mu)$ | $\Phi^{-1}(\eta)$ | *normalquantile* |
| $comp.log - log$ | $\log[-\log(-\mu)]$ | $1 - \exp[-\exp(\eta)]$ | $-$ |

And the table for canonical or default link, response range, and conditional variance function for GLM families.

| **Family** | **DefaultLink** | **Rangeofy** | **Var**$(y|\mathbf{x})$ |
|---|---|---|---|
| *gaussian* | *identity* | $(-\infty, +\infty)$ | $\phi$ |
| *binomial* | *logit* | $\frac{0,1,\ldots,N}{N}$ | $\frac{\mu(1-\mu)}{N}$ |
| *poisson* | *log* | $0, 1, \ldots$ | $\mu$ |
| *Gamma* | *inverse* | $(0, \infty)$ | $\phi\mu^2$ |
| *Inverse.gaussian* | $\frac{1}{\mu^2}$ | $(0, \infty)$ | $\phi\mu^3$ |

The variance distributions of an exponential family is a product of a positive *disperssion (scale)* parameter $\phi$ and a function of the mean given the linear predictor:

$$Var(y|\mathbf{x}) = \phi \times V[\mu(\mathbf{x})]$$

The variances for several exponential families are listed in the table above.

The *deviance*, based on the maximized value of the log-likelihood, provides a measure of the fit of a GLM to the data, much as the residual sum of squares does for a linear model. The value of the log-likelihood evaluated at the maximum likelihood estimates the regression coefficients for fixed dispersion is

$$\log L_0 = \sum \log p[y_i; \hat{\mu}(\mathbf{x_i}), \phi]$$

An fitting to a saturated model, with one parameter for each of the $n$ observations, with the mean response for each observation just the observed value

$$\log L_1 = \sum \log p[y_i; y_i, \phi]$$

THe resudual deviance is defined as twice the difference of the log-likelihoods,

$$D(\mathbf{y}; \hat{\mu}) = 2(\log L_1 - \log L_0)$$

the larger the diviance, the less well the the model of interest matches the data.

## GLMs for Binary Responce Data

Considering data in which each case provides a *binary response*, say “success” or “failure”, the cases are independent, and the probability of success $\mu(\mathbf{x})$ is the same for all cases with the same values $\mathbf{x}$ of the regressors.

When the response is binary, we think of the mean function $\mu(\mathbf{x})$ as the conditional probability that response is success given the values $\mathbf{x}$ of the regressors. The most common link funtction used with binary response data is the ligit link, for which

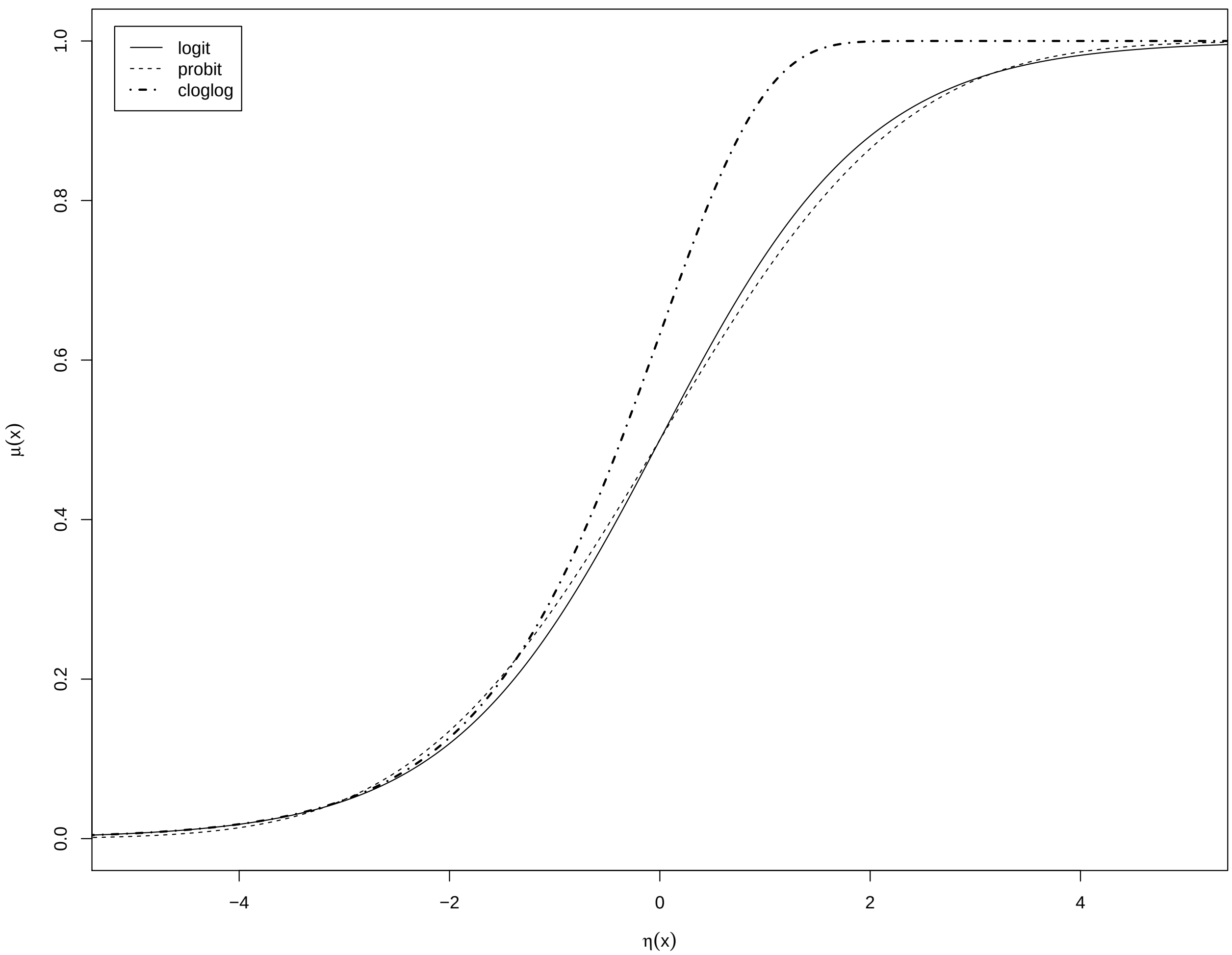
$$\log[\frac{\mu(\mathbf{x})}{1 - \mu(\mathbf{x})}] = \eta(\mathbf{x})$$

The left side of the equation is called the *logit* of the *log-odds*, where the *odds* are the probability of success divided by the probability of failure. Solving for $\mu(\mathbf{x})$ gives the mean function,

$$\mu(\mathbf{x}) = \frac{1}{1 + \exp[-\eta\mathbf{x})]}$$

Plot of the comparison of the logit, probit, and complementary log-log links

```
# Example from the car Book Chp 6
Probit <- binomial(link=probit)
Logit <- binomial(link=logit)
Cloglog <- binomial(link=cloglog)
range <- seq(-10,10,length=1000)
plot(range,Logit$linkinv(range),type="l", xlim=c(-5,5), lty=1,
       xlab=expression(eta(x)), ylab=expression(mu(x)))
lines(sqrt(pi^2/3)*range, Probit$linkinv(range), lty=2)
lines(range,Cloglog$linkinv(range), lty=4, lwd=2)
legend("topleft",c("logit", "probit", "cloglog"), lty=c(1,2,4),
       lwd=c(1,1,2), inset=0.02)
```



## Example: Women's Labor Force Participation

To illustrate logistic regression, we turn to a study of the U.S. Panel Study of Income Dynamics of the response variable is married women's labor force participation. The data is in Mroz (carData).

| Variable | Description | Remarcs |
|---|---|---|
| $lfp$ | labor force participation | $factor : no, yes$ |
| $k5$ | number of children ages 5 and younger | $0 - 3$ |
| $k618$ | number of children ages 6 to 18 | $0 - 8$ |
| $age$ | wife's age in yars | $30 - 60$ |
| $wc$ | wife's college attendance | $factor : no, yes$ |
| $hc$ | husband's college attendance | $factor : no, yes$ |
| $lwg$ | log of estimated wife's wage | |
| $inc$ | family income excluding wife's income | $1000s$ |

So, the estimated logistic-regression model is given by

$$log[\frac{\hat{\mu}(x)}{1-\hat{\mu}(x)}] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$$

If exponentiate both sides of the equation, we get

$$\frac{\hat{\mu}(x)}{1-\hat{\mu}(x)} = exp(\beta_0) \times exp(\beta_1 x_1) \times exp(\beta_2 x_2) \times \cdots \times exp(beta_k x_k)$$

where the left hand of the equation, $\frac{\hat{\mu}(x)}{1-\hat{\mu}(x)}$, gives the *fitted odds* of success, **the fitted probability of success divided by the fitted probability of failure**. Exponentiating the model removes the logarithms and changes the model in the log-odds scale to one that is multiplicative, in this log odds scale.

```
##################### From car
# carData Mroz
summary(Mroz)
```

```
  lfp          k5              k618             age             wc         hc     
 no :325   Min.   :0.0000   Min.   :0.000   Min.   :30.00   no :541   no :458  
 yes:428   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:36.00   yes:212   yes:295  
           Median :0.0000   Median :1.000   Median :43.00                      
           Mean   :0.2377   Mean   :1.353   Mean   :42.54                      
           3rd Qu.:0.0000   3rd Qu.:2.000   3rd Qu.:49.00                      
           Max.   :3.0000   Max.   :8.000   Max.   :60.00                      
      lwg               inc        
 Min.   :-2.0541   Min.   :-0.029  
 1st Qu.: 0.8181   1st Qu.:13.025  
 Median : 1.0684   Median :17.700  
 Mean   : 1.0971   Mean   :20.129  
 3rd Qu.: 1.3997   3rd Qu.:24.466  
 Max.   : 3.2189   Max.   :96.000  
```

```
# logistic model family= binomial's default link is logit
mroz.mod <- glm(lfp ~ k5 + k618 + age + wc + hc + lwg + inc, family=binomial, data=Mroz)
S(mroz.mod)
```

```
Call: glm(formula = lfp ~ k5 + k618 + age + wc + hc + lwg + inc, family =
          binomial, data = Mroz)

Coefficients:
```

```
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.182140   0.644375   4.938 7.88e-07 ***
k5          -1.462913   0.197001  -7.426 1.12e-13 ***
k618        -0.064571   0.068001  -0.950 0.342337
age         -0.062871   0.012783  -4.918 8.73e-07 ***
wcyes        0.807274   0.229980   3.510 0.000448 ***
hcyes        0.111734   0.206040   0.542 0.587618
lwg          0.604693   0.150818   4.009 6.09e-05 ***
inc         -0.034446   0.008208  -4.196 2.71e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1029.75  on 752  degrees of freedom
Residual deviance:  905.27  on 745  degrees of freedom

 logLik      df     AIC     BIC
-452.63       8  921.27  958.26

Number of Fisher Scoring iterations: 4

Exponentiated Coefficients and Confidence Bounds
              Estimate     2.5 %     97.5 %
(Intercept) 24.0982799 6.9377228 87.0347916
k5           0.2315607 0.1555331  0.3370675
k618         0.9374698 0.8200446  1.0710837
age          0.9390650 0.9154832  0.9625829
wcyes        2.2417880 1.4347543  3.5387571
hcyes        1.1182149 0.7467654  1.6766380
lwg          1.8306903 1.3689201  2.4768235
inc          0.9661401 0.9502809  0.9814042
```

Exponentiating the model removes the logarithms (S function shows the exponents of the betas). For example increasing the age of a woman by one year, holding the other predictors constant, from $odds = \exp(c_0) \times \exp(c_1) \times \exp(c_2) \times \exp(\beta_3) \times \exp(c_4) \ldots$ therefore multiplies the fitted odds of her being in the workforce by $\exp(\beta_3) = \exp(-0.06287) = 0.9391$. That is, reduces the odds of working by $100(1-0.9391) \approx 6\%$. Compared to a woman that who did not attend college, a college-educated woman with all other predictors fixed has fitted odds of working about 2.24 times higher, with a 95% confidence interval [1.43, 3.54]. The exponents of the coefficient estimates are called *risk factors* or *odds ratios*. The confidence intervals for the GLM are based on profiling the log-likelihood. The confidence intervals may not be symmetric.

## Volunteering for a Psychological Experiment

Cowles collected data on the willingness of students in an introductory psychology class to volunteer for a psychological experiment. The data set contains several variables: 1. The personality dimension *neuroticism*, a numeric variable with integer scores on a scale from zero to 24 2. The personality dimension *extraversion*, also a numeric variable with a potential range of zero to 24. 3. The factor sex, with levels "female" and "male". 4. Tha factor volunteer, with levels "no" and "yes".

Researchers expected volunteering to depend on the sex variable and on the interaction of the personality dimensions, so included the linear-by-linear interaction between nuroticism and extraversion:

```r
brief(Cowles)
```

```
1421 x 4 data.frame (1416 rows omitted)
```

```
     neuroticism extraversion    sex volunteer
             [i]          [i]    [f]       [f]
1             16           13 female        no
2              8           14 male          no
3              5           16 male          no
. . .
1420          19           20 female       yes
1421          15           20 male         yes
```

```r
sum(Cowles$volunteer == "yes") # number yes
```

```
[1] 597
```

```r
cowles.mod <- glm(volunteer ~ sex + neuroticism*extraversion,
    data=Cowles, family=binomial)
brief(cowles.mod, pvalues=TRUE)
```

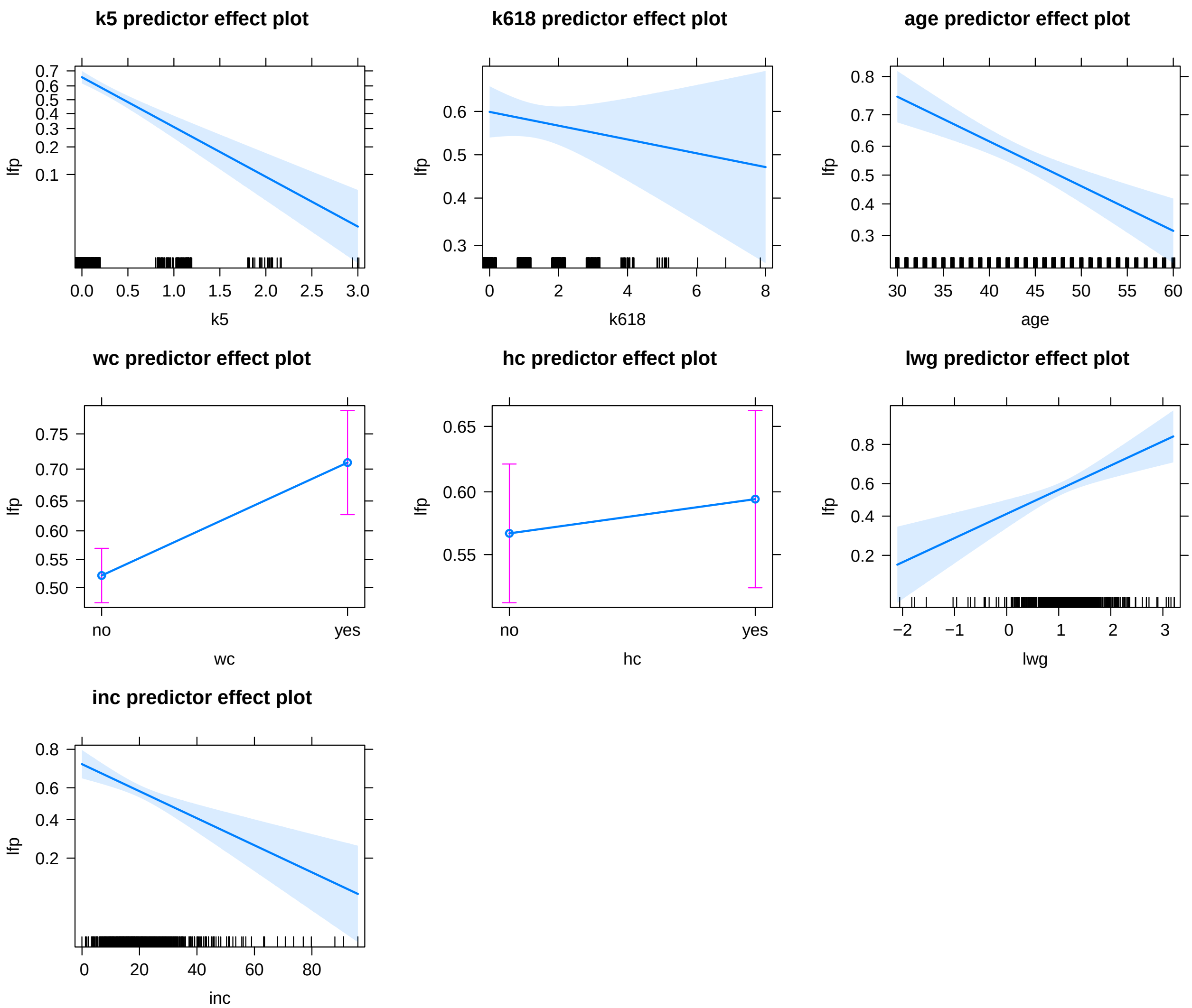
```
              (Intercept) sexmale neuroticism extraversion
Estimate        -2.36e+00 -0.2472     0.11078     1.67e-01
Std. Error       5.01e-01  0.1116     0.03765     3.77e-02
Pr(>|t|)         2.55e-06  0.0268     0.00326     9.75e-06
exp(Estimate)    9.46e-02  0.7810     1.11715     1.18e+00
              neuroticism:extraversion
Estimate                      -0.00855
Std. Error                     0.00293
Pr(>|t|)                       0.00355
exp(Estimate)                  0.99148

 Residual deviance = 1897 on 1416 df
```

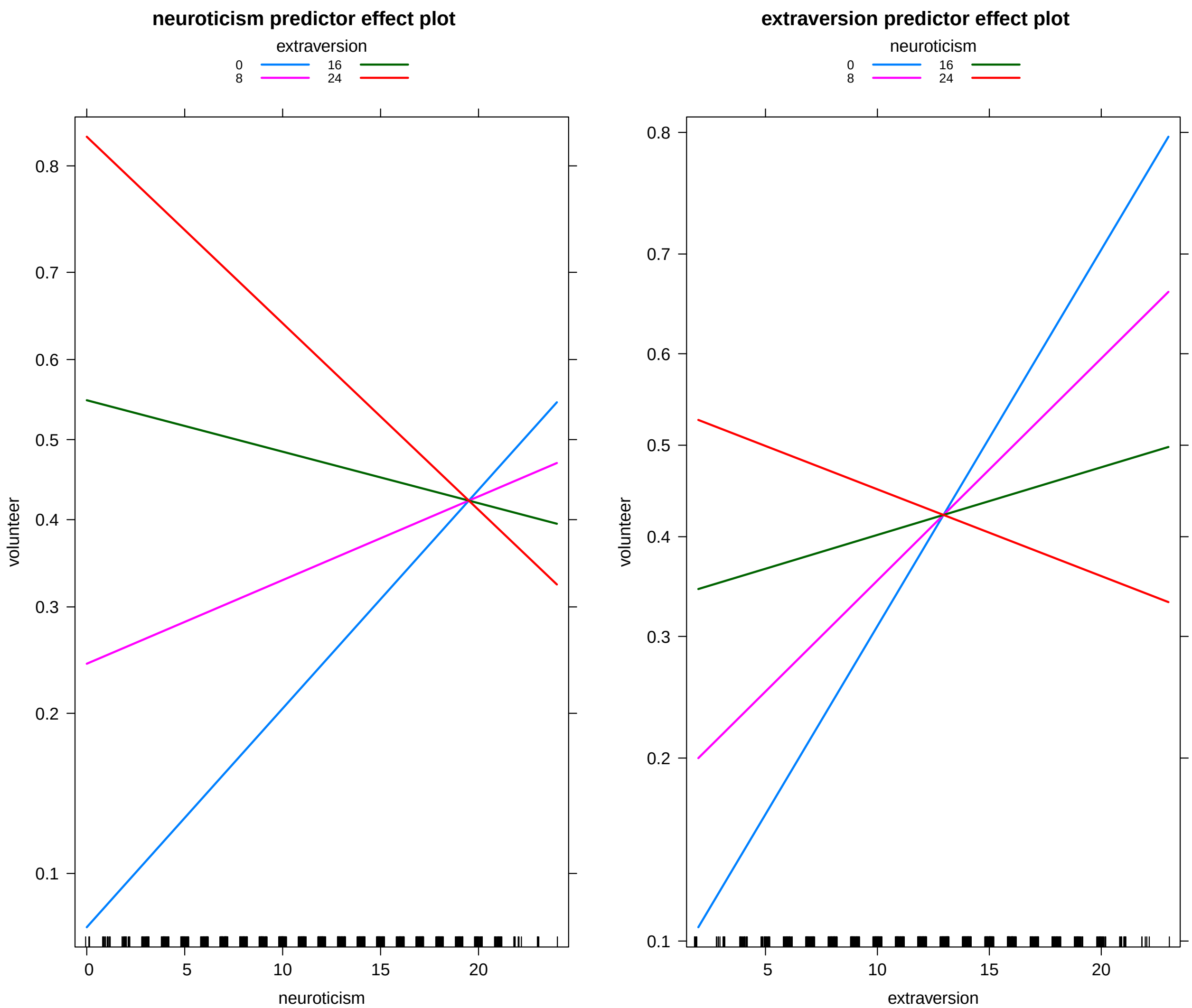
## Predictor Effect Plots for Logistic Regression

The **effects** package, can draw predictor effect plots for generalized linear models, including logistic regression.

```r
plot(predictorEffects(mroz.mod))
```




```
plot(predictorEffects(cowles.mod, ~ neuroticism + extraversion,
    xlevels=list(neuroticism=seq(0, 24, by=8),
        extraversion=seq(0, 24, by=8))),
    lines=list(multiline=TRUE))
```




## Analysis of Deviance and Hypotesis Test for Logistic Regression

### Model Comparisons

```
mroz.mod.2 <- update(mroz.mod, . ~ . - k5 - k618)
anova(mroz.mod.2, mroz.mod, test="Chisq")
```

```
Analysis of Deviance Table

Model 1: lfp ~ age + wc + hc + lwg + inc
Model 2: lfp ~ k5 + k618 + age + wc + hc + lwg + inc
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       747     971.75
2       745     905.27  2   66.485 3.655e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
brief(cowles.mod.0 <- update(cowles.mod,
    . ~ . - neuroticism:extraversion))
```

```
              (Intercept) sexmale neuroticism extraversion
```

```
Estimate            -1.116  -0.235     0.00636       0.0663
Std. Error           0.249   0.111     0.01136       0.0143
exp(Estimate)        0.327   0.790     1.00638       1.0686

 Residual deviance = 1906 on 1417 df
```

```r
anova(cowles.mod.0, cowles.mod, test="Chisq")
```

```
Analysis of Deviance Table

Model 1: volunteer ~ sex + neuroticism + extraversion
Model 2: volunteer ~ sex + neuroticism * extraversion
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      1417     1906.1
2      1416     1897.4  1   8.6213 0.003323 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Type II Tests
Anova(mroz.mod)
```

```
Analysis of Deviance Table (Type II tests)

Response: lfp
     LR Chisq Df Pr(>Chisq)
k5     66.484  1  3.527e-16 ***
k618    0.903  1   0.342042
age    25.598  1  4.204e-07 ***
wc     12.724  1   0.000361 ***
hc      0.294  1   0.587489
lwg    17.001  1  3.736e-05 ***
inc    19.504  1  1.004e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
Anova(cowles.mod)
```

```
Analysis of Deviance Table (Type II tests)

Response: volunteer
                         LR Chisq Df Pr(>Chisq)
sex                        4.9184  1   0.026572 *
neuroticism                0.3139  1   0.575316
extraversion              22.1372  1  2.538e-06 ***
neuroticism:extraversion   8.6213  1   0.003323 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Other Hypothesis Tests
linearHypothesis(mroz.mod, c("k5", "k618"))
```

```
Linear hypothesis test

Hypothesis:
k5 = 0
k618 = 0
```

```
Model 1: restricted model
Model 2: lfp ~ k5 + k618 + age + wc + hc + lwg + inc

  Res.Df Df  Chisq Pr(>Chisq)
1    747
2    745  2 55.163  1.051e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linearHypothesis(mroz.mod, "k5 = k618")
```

```
Linear hypothesis test

Hypothesis:
k5 - k618 = 0

Model 1: restricted model
Model 2: lfp ~ k5 + k618 + age + wc + hc + lwg + inc

  Res.Df Df  Chisq Pr(>Chisq)
1    746
2    745  1 49.479  2.005e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Fitted and Predicted Values

The function `predict()` returns the estimated linear predictor values for each observation. And to get the fitted probabilities we use the argument type="response". The `fitted()` function can also be used.

```
opts <- options(digits=5)
head(predict(mroz.mod)) # first few values
```

```
        1         2         3         4         5         6
 0.063337  0.693821 -0.174106  0.672295  0.677721  0.388719
```

```
head(predict(mroz.mod, type="response"))
```

```
      1       2       3       4       5       6
0.51583 0.66682 0.45658 0.66202 0.66323 0.59597
```

And the `predict()` can be used to compute predicted values for arbitrary combinations of predictor values. For example we can estimate the probability of volunteering at neuroticism and extraversion at 12.

```
options(opts)

predict.data <- data.frame(sex=c("female", "male"),
    neuroticism=rep(12, 2), extraversion=rep(12, 2))
predict.data$p.volunteer <- predict(cowles.mod,
    newdata=predict.data, type="response")
predict.data
```

```
     sex neuroticism extraversion p.volunteer
1 female          12           12   0.4356968
2   male          12           12   0.3761793
```

## Binomial Data

In binomial response data, the response variable $y_i$ for each case $i$ is the number of successes in a fixed number $N_i$ of independent trials, each with the same probability of success. Binary regression is a limiting case of binomial regression with all the $N_i = 1$

| Perceived Closeness | Intensity of Preference | Voted | Did Not Voted | **logit** |
|---|---|---|---|---|
| *One − sided* | Weak | 91 | 39 | 0.847 |
| *One − sided* | Medium | 121 | 49 | 0.904 |
| *One − sided* | Strong | 64 | 24 | 0.981 |
| *Close* | Weak | 214 | 87 | 0.900 |
| *Close* | Medium | 284 | 76 | 1.318 |
| *Close* | Strong | 201 | 25 | 2.084 |

```r
Campbell <- data.frame(
    closeness = factor(rep(c("one.sided", "close"), c(3, 3)),
        levels=c("one.sided", "close")),
    preference = factor(rep(c("weak", "medium", "strong"), 2),
        levels=c("weak", "medium", "strong")),
    voted = c(91, 121, 64, 214, 284, 201),
    did.not.vote = c(39, 49, 24, 87, 76, 25)
)
Campbell
```

```
  closeness preference voted did.not.vote
1 one.sided       weak    91           39
2 one.sided     medium   121           49
3 one.sided     strong    64           24
4     close       weak   214           87
5     close     medium   284           76
6     close     strong   201           25
```

For binomial data, the response can be the *proportion* of successes for each observation, or the proportion of successes for failures. For example the logit for the One-sided, weak case 91 voted, and 39 did not voted, log(`Voted/Did not Voted`) = log(91/39) -> *0.847*

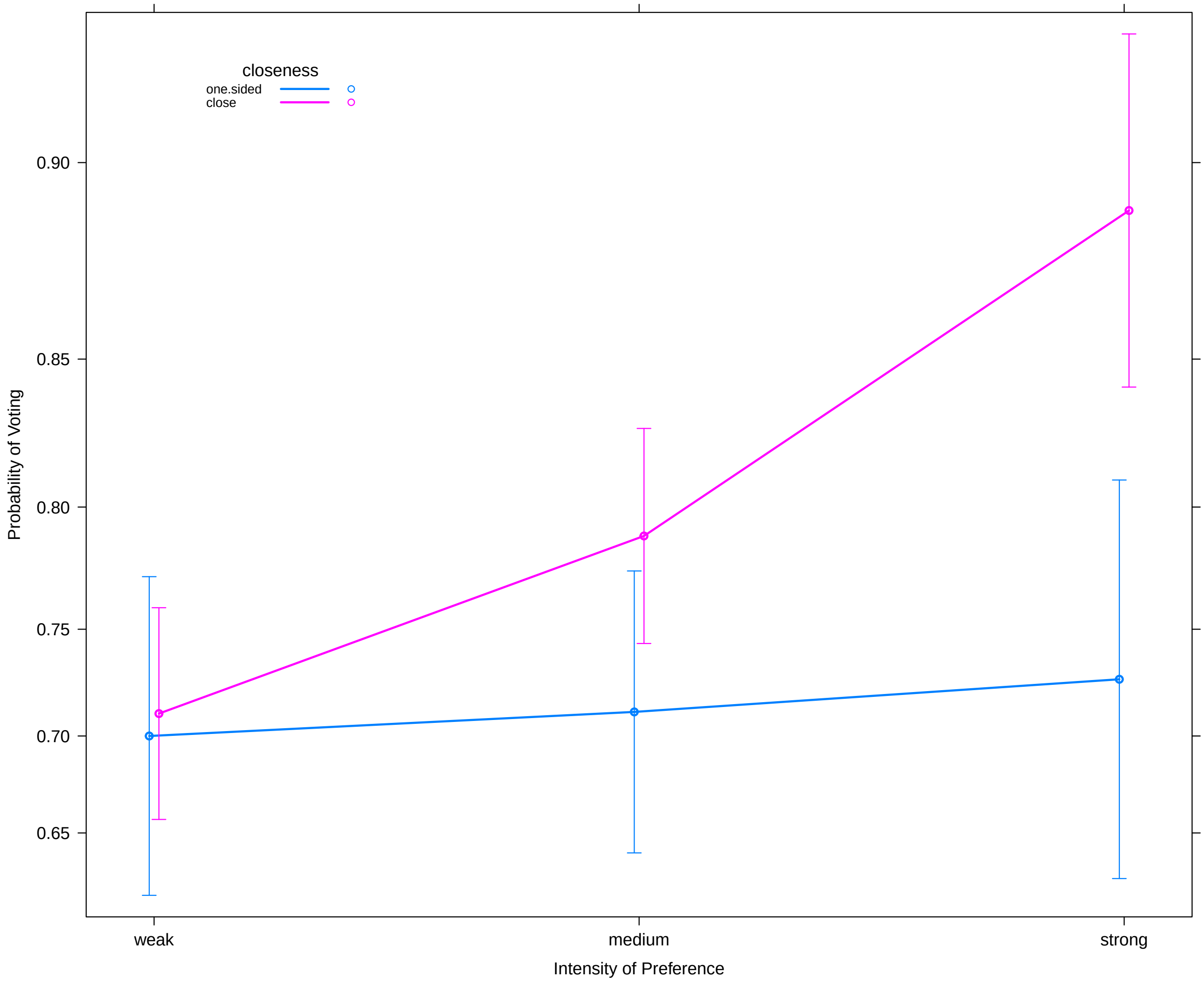
```r
campbell.mod <- glm(cbind(voted, did.not.vote) ~
    closeness*preference, family=binomial, data=Campbell)
# the estimated responses are exact and the residuals are zero
predict(campbell.mod)
```

```
        1         2         3         4         5         6
0.8472979 0.9039702 0.9808293 0.9000679 1.3182409 2.0844291
```

```r
residuals(campbell.mod)
```

```
[1] 0 0 0 0 0 0
```

```r
plot(predictorEffects(campbell.mod, ~ preference),
     main="", confint=list(style="bars"), lines=list(multiline=TRUE),
     xlab="Intensity of Preference", ylab="Probability of Voting",
     lattice=list(key.args = list(x =0.1, y = 0.95, corner=c(0, 1))))
```




The emmeans function can be used to test differences between the two levels of closeness fo each level of preference:

```
emmeans(campbell.mod, pairwise ~ closeness | preference)$contrasts
```

```
preference = weak:
 contrast          estimate    SE  df z.ratio p.value
 one.sided - close  -0.0528 0.230 Inf -0.230  0.8184

preference = medium:
 contrast          estimate    SE  df z.ratio p.value
 one.sided - close  -0.4143 0.213 Inf -1.945  0.0517

preference = strong:
 contrast          estimate    SE  df z.ratio p.value
 one.sided - close  -1.1036 0.320 Inf -3.451  0.0006

Results are given on the log odds ratio (not the response) scale.
```