Statistical Thinking using Randomisation and Simulation

Generalised Linear Models

Di Cook University

W6.C2

Generalised linear models

- Overview
- Types
- Assumptions
- Fitting
- Examples

Overview

- GLMs are a broad class of models for fitting different types of response variables distributions.
- The multiple linear regression model is a special case.

Three components

- Random Component: probability distribution of the response variable
- Systematic Component: explanatory variables
- Link function: describes the relaionship between the random and systematic components

Multiple linear regression

- Random component: has a normal distribution, and so
- Systematic component:
- Link function: identity, just the systematic component

Poisson regression

- takes integer values, 0, 1, 2, …
- Link function: , name=`log`. (Think of as .)

Bernouilli, binomial regression

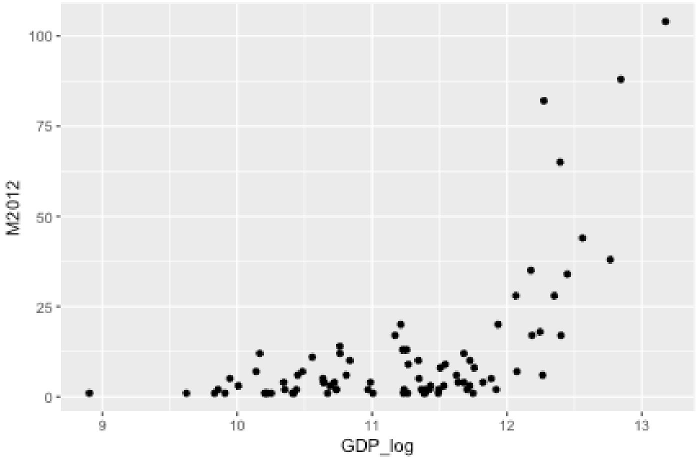
- takes integer values, (bernouilli), (binomial)
- Let , link function is , name=`logit`

Assumptions

- The data are independently distributed, i.e., cases are independent.
- The dependent variable does NOT need to be normally distributed, but it typically assumes a distribution from an exponential family (e.g. binomial, Poisson, multinomial, normal,...)
- Linear relationship between the transformed response (see examples below)
- Explanatory variables can be transformations of original variables
- Homogeneity of variance does NOT need to be satisfied for original units, but it should be still true on the transformed response scale
- Uses maximum likelihood estimation (MLE) to estimate the parameters
- Goodness-of-fit measures rely on sufficiently large samples

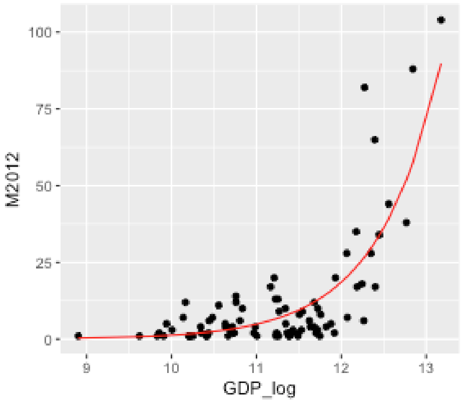
Example: Olympics medal tally

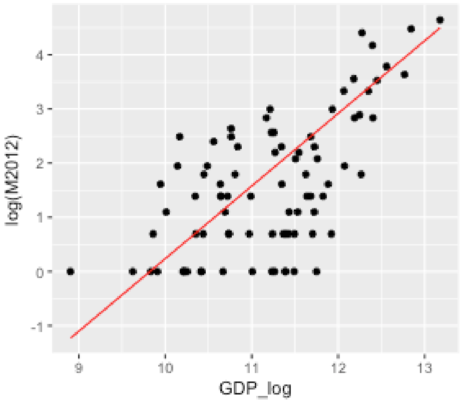
- Model medal counts on log_GDP
- Medal counts = integer, which suggests using a Poisson model.



Model fit and what it looks like

```
oly_glm <- glm(M2012~GDP_log, data=oly_gdp2012,
               family=poisson(link=log))
summary(oly_glm)$coefficients
#>             Estimate Std. Error z value Pr(>|z|)
#> (Intercept)    -13.2      0.538     -24 3.6e-132
#> GDP_log          1.3      0.045      30 6.8e-198
```





Your turn

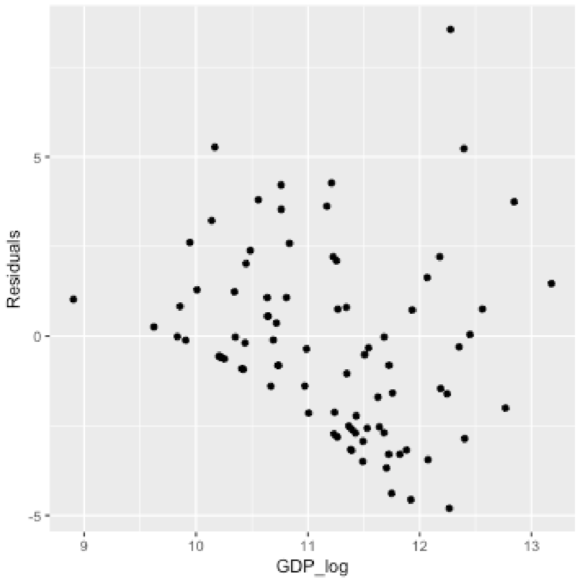
Write down the formula of the fitted model.

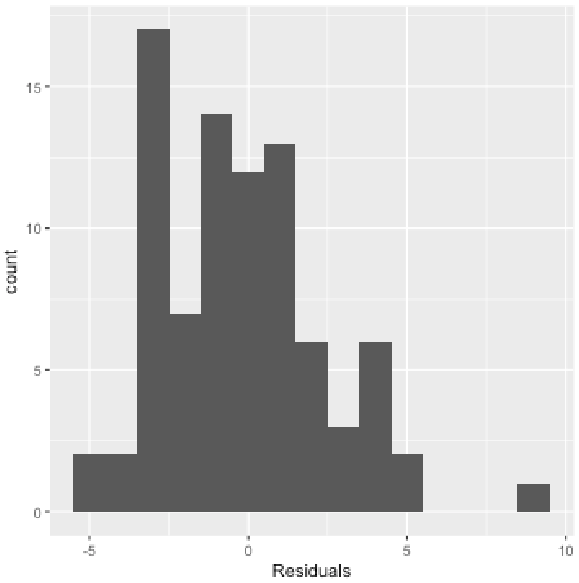
Model fit

```
#>
#> Call:
#> glm(formula = M2012 ~ GDP_log, family = poisson(link = log),
#>     data = oly_gdp2012)
#>
#> Deviance Residuals:
#>    Min      1Q  Median      3Q     Max
#>  -4.80   -2.22   -0.36    1.07    8.55
#>
#> Coefficients:
#>             Estimate Std. Error z value Pr(>|z|)
#> (Intercept) -13.1691     0.5383   -24.5   <2e-16 ***
#> GDP_log       1.3406     0.0447    30.0   <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> (Dispersion parameter for poisson family taken to be 1)
#>
#>     Null deviance: 1567.70  on 84  degrees of freedom
#> Residual deviance:  545.92  on 83  degrees of freedom
#> AIC: 845.7
#>
#> Number of Fisher Scoring iterations: 5
```

The difference between the null and residual deviance is substantial, suggesting a good fit.

Residual plots





Heteroskedasticity in residuals. One fairly large residual.

Influence statistics

```
#>          .rownames .cooksd .resid
#> 1        RussianFed 1.9e+00  8.553
#> 2             China 1.5e+00  3.743
#> 3      UnitedStates 8.3e-01  1.468
#> 4      GreatBritain 8.0e-01  5.232
#> 5           Jamaica 4.4e-01  5.267
#> 6             India 2.6e-01 -4.800
#> 7             Japan 2.5e-01 -2.010
#> 8              Cuba 2.4e-01  4.215
#> 9           Ukraine 2.3e-01  4.270
#> 10            Kenya 1.9e-01  3.802
#> 11          Belarus 1.6e-01  3.535
#> 12          Hungary 1.5e-01  3.621
#> 13           Brazil 1.5e-01 -2.862
#> 14          Georgia 1.3e-01  3.219
#> 15        Indonesia 1.2e-01 -4.563
#> 16           Mexico 9.8e-02 -3.444
#> 17      SaudiArabia 9.2e-02 -4.388
#> 18        Australia 7.6e-02  2.211
#> 19       Azerbaijan 7.5e-02  2.584
#> 20         Mongolia 7.3e-02  2.612
#> 21    ChineseTaipei 7.0e-02 -3.680
#> 22           Turkey 6.5e-02 -3.179
#> 23      Switzerland 6.5e-02 -3.293
#> 24         Ethiopia 6.2e-02  2.385
#> 25          Belgium 6.0e-02 -3.294
#> 26        Venezuela 5.8e-02  3.498
```

Prediction from the model

```
aus <- oly_gdp2012 %>% filter(Code == "AUS")
predict(oly_glm, aus)
#>   1
#> 3.2
```

WAIT! What??? Australia earned more than 3 medals in 2012. Either the model is terrible, or we've made a mistake!

Prediction from the model

```
aus <- oly_gdp2012 %>% filter(Code == "AUS")
predict(oly_glm, aus)
#>    1
#> 3.2
```

WAIT! What??? Australia earned more than 3 medals in 2012. Either the model is terrible, or we've made a mistake!

```
aus <- oly_gdp2012 %>% filter(Code == "AUS")
predict(oly_glm, aus, type="response")
#>  1
#> 23
```

Prediction from the model

```
aus <- oly_gdp2012 %>% filter(Code == "AUS")
predict(oly_glm, aus)
#>    1
#> 3.2
```

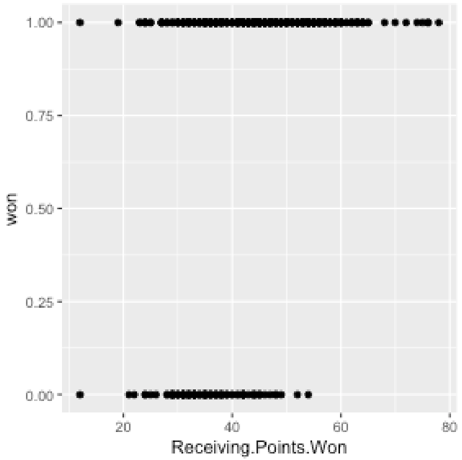
WAIT! What??? Australia earned more than 3 medals in 2012. Either the model is terrible, or we've made a mistake!

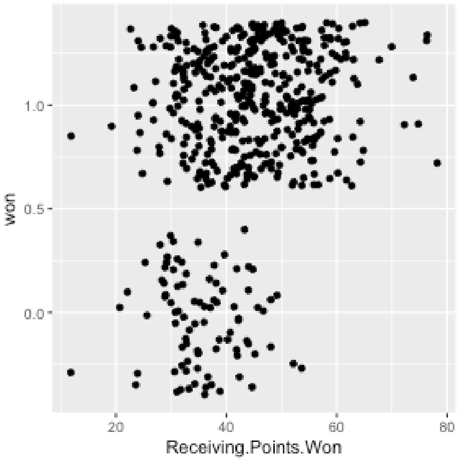
```
aus <- oly_gdp2012 %>% filter(Code == "AUS")
predict(oly_glm, aus, type="response")
#>  1
#> 23
```

Need to transform predictions into original units.

Example: winning tennis matches

We have data scraped from the web sites of the 2012 Grand Slam tennis tournaments. There are a lot of statistics on matches. Below we have the number of receiving points won, and whether the match was won or not.





Your turn

The response variable is binary. What type of GLM should be fit?

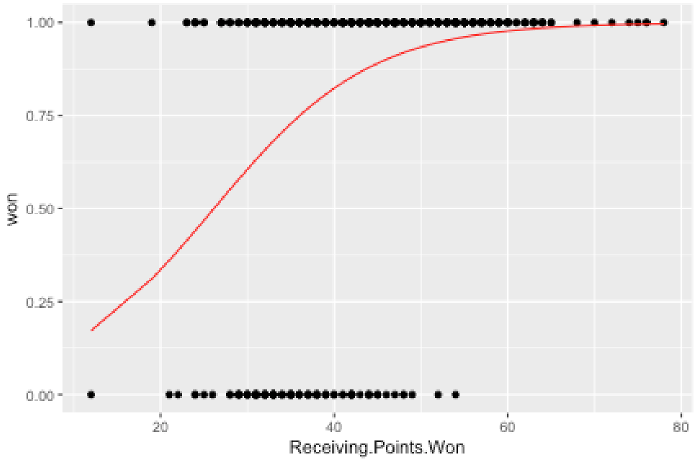
Your turn

The response variable is binary. What type of GLM should be fit?
bernouilli/binomial

Model

```
tennis_glm <- glm(won~Receiving.Points.Won, data=tennis,
                  family=binomial(link='logit'))
```

```
#>                          Estimate Std. Error z value Pr(>|z|)
#> (Intercept)                 -2.91      0.586    -5.0  7.1e-07
#> Receiving.Points.Won         0.11      0.015     7.3  3.0e-13
```



Your turn

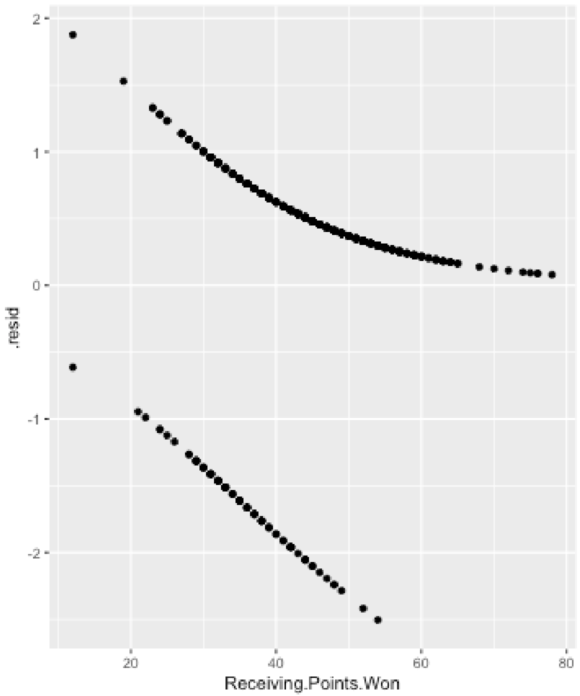
Write down the fitted model

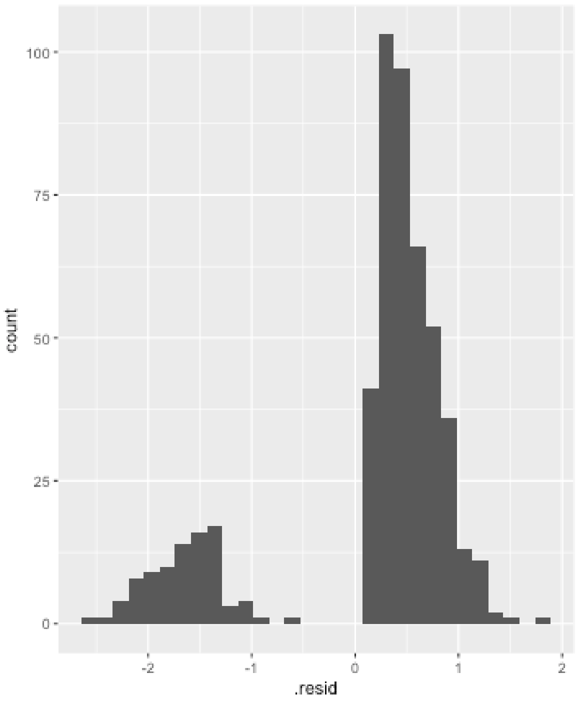
Model fit

```
#>
#> Call:
#> glm(formula = won ~ Receiving.Points.Won, family = binomial(link = "logit"),
#>     data = tennis)
#>
#> Deviance Residuals:
#>    Min       1Q   Median       3Q      Max
#> -2.506    0.227    0.411    0.624    1.877
#>
#> Coefficients:
#>                      Estimate Std. Error z value Pr(>|z|)
#> (Intercept)           -2.9053     0.5860   -4.96  7.1e-07 ***
#> Receiving.Points.Won   0.1111     0.0152    7.29  3.0e-13 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> (Dispersion parameter for binomial family taken to be 1)
#>
#>     Null deviance: 472.99  on 511  degrees of freedom
#> Residual deviance: 402.16  on 510  degrees of freedom
#> AIC: 406.2
#>
#> Number of Fisher Scoring iterations: 5
```

Not much difference between null and residual deviance, suggests return points won does not explain much of the match result.

Residuals





Model is just not capturing the data very well. There are two groups of residuals, its overfitting a chunk and underfitting chunks of data.

Influence statistics

```
#>     .cooksd .resid
#> 1   6.0e-02  1.877
#> 2   3.6e-02 -2.505
#> 3   2.9e-02 -2.420
#> 4   2.4e-02  1.528
#> 5   2.0e-02 -2.287
#> 6   1.7e-02 -2.242
#> 7   1.7e-02 -2.242
#> 8   1.5e-02 -2.196
#> 9   1.3e-02 -2.149
#> 10  1.2e-02  1.329
#> 11  1.2e-02  1.329
#> 12  1.1e-02 -2.103
#> 13  1.1e-02 -2.103
#> 14  1.1e-02 -2.103
#> 15  9.9e-03 -2.055
#> 16  9.9e-03 -2.055
#> 17  9.9e-03 -2.055
#> 18  9.9e-03 -2.055
#> 19  9.4e-03  1.280
#> 20  9.4e-03  1.280
#> 21  9.4e-03  1.280
#> 22  9.4e-03  1.280
#> 23  8.6e-03 -2.008
#> 24  7.6e-03  1.232
#> 25  7.6e-03  1.232
#> 26  7.5e-03  1.959
```

Prediction from the model

```r
newdata <- data.frame(Receiving.Points.Won=c(20, 50), won=c(NA, NA))
predict(tennis_glm, newdata, type="response")
#>    1    2
#> 0.34 0.93
```

Interpret the response as the probability of winning if your receiving points was 20, 50.

Summary

Generalised linear models are a systematic way to fit different types of response distributions.

Resources

- Beginners guide
- Introduction to GLMs
- Quick-R GLMs
- The Analysis Factor, Generalized Linear Models Parts 1-4
- wikipedia
- Do Smashes Win Matches?

Share and share alike