Introduction to $\mathcal{R}$

Session 6: GLMs

Dag Tanneberg[1]

Potsdam Center for Quantitative Research
University of Potsdam, Germany

October 11/12, 2018

[1]Chair of Comparative Politics, UP, dag.tanneberg@uni-potsdam.de

Introduction

Before we start. . .

- Quit & reopen RStudio.
- Load "./06/dta/asoiaf.csv"" from the course material.
  - **Remember**: Uncheck the option "Strings as factors"
- Open a new script file.
- Execute the code below. What does it do?

```
asoiaf[, "died"] <- !is.na(asoiaf[, "book_of_death"])
```

What do we intent to do?

- **Question**: What's the chance that Jon Snow is going to die?
- **Means**: Regression on a linear combination of predictors

$$p(Death = 1|x, \beta) = \sum_K \beta_k x_k$$

- **Problem**: Chance of death is not a well-behaved response.
  a. We don't obseverve probabilities but discrete events.
  b. Probabilities are restricted to $[0, 1]$, but $\mathbf{X}\beta$ can take any value.

Some Intuition on GLMs

- **Challenge**: Map the linear combination $\mathbf{X}\beta$ into a domain which fits our response.
- Applies to many quantities of interest, e.g.,
  - Household income
  - Satisfaction with democracy
  - Number of bills per session of parliament
  - ...
- GLMs: link function $g(\cdot)$ relates response to linear predictor $\mathbf{X}\beta$
  - logit transformation $[ln(\frac{p}{1-p})]$ for binary DVs
  - natural log $(ln(\mu))$ for count data

Outline

The Basics of Running GLMs in $\mathcal{R}$

Generic Format of Fitting GLMs

```
fit <- glm(
  formula = <formula>,
  family = <family>(link = "<link>"),
  data = <data>,
  weights = <weights>, # Be careful! Meaning changes
                       # depending on <family>.
  subset = <subset>,
  na.action = na.omit, # Retains only complete cases.
  <...> # Options to tweak the optimizer.
)
```

$\mathcal{R}$'s Formula Interface[2]

Generic Example

$$y \sim x_1 + x_2 + \cdots + x_k$$

Formula Creation

Symbol	Meaning	Example
:	Specify an interaction	$y \sim x : z \Rightarrow y = xz$
$*$	Specify all possible interactions	$y \sim x * z \Rightarrow y = x + z + xz$
$\hat{}$	Specify interactions up to some degree	$y \sim (x+z)^2 \Rightarrow y = x + z + xz$
.	Wildcard for all other variables	$y \sim . \Rightarrow y = x + z + w + \ldots$
-	Remove variable(s)	$y \sim (x+z)^{2\smile} x : z \Rightarrow y = x + z$
-1 OR $0+$	Remove the intercept	$y \sim x - 1$ OR $y \sim 0 + x$
$I()$	Arithmetical transformation	$y \sim I(x^2) \Rightarrow y = x^2$
function	Other mathematical transformations	log10$(y) \sim x \Rightarrow log_{10}(y) = x$

[2] Adapted from Kabacoff, R. 2011. R in Action. Shelter Island: Manning Publications, p. 178.

$\mathcal{R}$'s Formula Interface, contd.

Exercise How would you write the following formulas?[3]

1. $y = a + x + z + xz$
2. $y = a + x + x^2 + x^3$
3. $log_e(y) = x + z + w + xz + xw + wz$
4. y as a function of variables in the data but k

[3] Assume a is the constant.

Family Generators and Link Functions in glm()[4]

A Practical Example

```
glm(<...>, family = binomial(link = "logit"), <...>)
```

	link = "<arg>"							
family	μ identity	μ^{-1} inverse	$ln(\mu)$ log	$ln(\frac{\mu}{1-\mu})$ logit	$\Phi(\mu)$ probit	$ln[-ln(1-\mu)]$ cloglog	$\sqrt{\mu}$ sqrt	$\frac{1}{\mu^2}$ 1/mu^2
gaussian()	•	○	○					
binomial()			○	•	○	○		
poisson()	○		•				○	
Gamma()	○	•	○					
inverse.gaussian()	○	○	○					•
quasi()	•	○	○	○	○	○	○	○
quasibinomial()				•	○	○		
quasi()	○		•				○	

Legend: • default, ○ possible

[4] Adapted from Fox, J. and S. Weisberg. 2011. An R Companion to Applied Regression. 2nd ed. London: SAGE, pp. 231, 233.

Get Your Hands Dirty

Now it's your turn. Use the **asoiaf** data to

- regress **died** on
- **allegiances**,
- the full interaction of **gender** and **nobility**, and
- a cubic polynomial on **age_in_chapters**.
- This should be a **logistic** regression model.
- Save the results to an object called **myfit**.

Solution to the Exercise

```r
myfit <- glm(
  formula = died ~ 0 + allegiances +
    gender * nobility +
    age_in_chapters + I(age_in_chapters^2) +
    I(age_in_chapters^3),
  family = binomial(link = "logit"),
  data = asoiaf
)
```

Working With Regression Results

A Menu of Typical Options[5]

Function	Output
summary()	Display detailed model results
coef()	Display fitted model parameters
confint()	Provide confidence intervals
fitted()	Return fitted values
residuals()	Return residual values
anova()	Return an ANOVA table for a fitted model or compare fitted models
vcov()	Return the variance-covariance matrix
AIC()	Return Akaike's Information Criterion
plot()	Display diagnostics plots
predict()	Predict response values for new data

[5] Adapted from Kabacoff, R. 2011. R in Action. Shelter Island: Manning Publications, p. 179.

How to Predict New Data

Generic Sequence

1. Define scenarios to predict
2. Create a date frame which contains those scenarios
3. Use predict() to return quantities of interest
4. Summarize the results

Let's Do One Example Together

```
# Steps 1 & 2
pred_dta <- data.frame(
  allegiances = "Baratheon",
  gender = mean(asoiaf$gender),
  nobility = mean(asoiaf$nobility),
  age_in_chapters = 0:343, stringsAsFactors = FALSE
)
# Step 3
pred_dta[, "fitted"] <- predict(
  myfit, newdata = pred_dta, type = "response"
)
# Step 4
ggplot(data = pred_dta,
  aes(x = age_in_chapters, y = fitted)) + geom_line()
```

Get Your Hands Dirty

Now it's your turn. Is John Snow going to die? Setup possible scenarios and evaluate the results.

One Possible Solution

```
jon_snow <- which(asoiaf$name == "Jon Snow")
pred_dta <- asoiaf[rep(jon_snow, 3), ]; rm(jon_snow)
pred_dta[2, "allegiances"] <- "Stark"
pred_dta[3, "allegiances"] <- "Targaryen"
pred_dta[, "fitted"] <- predict(
  myfit, newdata = pred_dta, type = "response"
)
pred_dta[, "fitted"]
```

Checking Assumptions

Checking Assumptions

- Always check your diagnostic plots

```
plot(myfit)
```

- For detailed instructions see: Fox, J. and S. Weisberg. 2011. An R Companion to Applied Regression. 2nd ed. London: SAGE.

Summary

Summary

- base $\mathcal{R}$ offers many probability models
- Numerous extensions are available (see the CRAN Taskviews)
- Discussion of marginal effects requires some acrobatics