Different models for different purposes: focused model comparison in R

Chris Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

Abstract

This is a brief tutorial in the "focused" model comparison methods developed by Claeskens and Hjort.

These can be implemented in the 'fic' R package.

The method originates from the paper by Claeskens and Hjort (2003) , and is developed further in the book by Claeskens and Hjort (2008) and a series of related papers.

TODO make abstract more exciting

Keywords: FIC,model comparison,AIC,BIC.

1. Introduction: principles for model comparison

To compare a set of statistical models fitted to the same data by maximum likelihood, it is common to rank them according to some "criterion". For example, Akaike's information criterion (AIC, Akaike (1973)) takes the form

$$-2\log \ell(\hat{\theta};\mathbf{x}) + 2p$$

where $\ell(\hat{\theta};\mathbf{x})$ is the maximised likelihood for the model fitted to the dataset $\mathbf{x}$, the likelihood is maximised at parameters $\hat{\theta}$, and p is the number of parameters.

The Bayesian information criterion (BIC, Schwarz (1978)) is

$$-2\log \ell(\hat{\theta};\mathbf{x}) + p\log(n)$$

These two criteria are based on very different principles. Thus they often rank models differently. The AIC is designed to choose the model with the best predictive ability, thus it tends to favour bigger models as the sample size increases. BIC is an approximation to Bayesian model comparison by Bayes factors, and selects the model with the highest posterior probability under an implicit weak prior (with an amount of information equivalent to one observation, see Kass and Wasserman (1995)). If there is a "true" model, the BIC will select it "consistently" as the sample size increases. In many situations there may not be a true model, and collecting more data will uncover more complexity in the process generating the data, in which case AIC may be more suitable. See e.g. Burnham and Anderson (2003), Claeskens and Hjort (2008) for more theory behind these criteria.

However both of these methods select one "best fitting" model for a given dataset. But that might not always be appropriate. Different models may be better for different purposes. This is the idea behind "focused" model comparison.

[describe what package does, structure of paper]

2. Focused model comparison: principles and formulae.

Suppose the range of models we are willing to use is bounded by

- a *wide model*, in which all models we would use are nested, with parameters $(\boldsymbol{\theta}, \boldsymbol{\gamma})$
- a *narrow model*, the smallest model we are willing to use, defined by setting $\boldsymbol{\gamma} = \boldsymbol{\gamma}_0$ in the wide model.

`[list of examples. covariate selection obvious, but lots more too]`

Suppose also that the purpose of the model is to estimate some *focus* quantity, which could be any function of the basic parameters

$$\mu = g(\boldsymbol{\theta}, \boldsymbol{\gamma})$$

In focused model comparison, we prefer models which give better estimates of μ. A typical way to define "better" is by the *mean square error*. The mean square error of the estimate $\hat{\mu}_S$ under a submodel S of the wide model, compared to the true value μ, is

$$E\left\{(\hat{\mu}_S - \mu)^2\right\}$$

This expectation is calculated under the assumption that the data are generated from the wide model. While we believe the wide model is the most realistic, we also accept that there may not be enough data to give sufficiently precise estimates of μ. Therefore we are willing to accept some bias in this estimate, in return for a smaller variance, by selecting a smaller model than the wide model. The submodel S with the lowest mean square error is the one which makes the optimal trade-off between bias and variance.

The expected squared error under model S can be decomposed as a sum of the squared bias B^2 and the variance V.

$$\begin{aligned} MSE = E\left\{(\hat{\mu}_S - \mu)^2\right\} &= (E(\hat{\mu}_S) - \mu)^2 + E\left\{(\hat{\mu}_S - E(\hat{\mu}_S))^2\right\} \\ &= B^2 + V \end{aligned} \tag{1}$$

Estimators for these quantities are constructed by Claeskens and Hjort (2003) under an asymptotic framework in which the data are assumed to be n independent observations generated from from the wide model, but reparameterised so that $\boldsymbol{\gamma} = \gamma_0 + \boldsymbol{\delta}/\sqrt{n}$. Thus as the sample size increases, we aim to detect more subtle departures from the narrow model.

[any more basic assumptions?]

An obvious estimator for the bias B is $\hat{B} = \hat{\mu_S} - \hat{\mu_W}$, where $\hat{\mu}_W$ is the estimate of the focus quantity under the wide model, which is assumed to be unbiased. However, Claeskens and Hjort (2003) derive a more accurate estimator for the *squared* bias as

$$\widehat{B^2} = (\psi_{full} - \psi_S)^2/n,$$

where

- $\psi_{full} = \hat{\omega}^T \hat{\boldsymbol{\delta}}$ and $\psi_S = \hat{\omega}^T G_S \hat{\boldsymbol{\delta}}$ are estimates of $\omega^T \boldsymbol{\delta}$ under the wide model and submodel respectively, with $G_S = \pi^T Q_S \pi Q^{-1}$ *[intuitive explanation?]*
- $\hat{\gamma}$ is the estimate of γ under the wide model, and $\hat{\delta} = \hat{\gamma}\sqrt{n}$,
- $\hat{\omega} = J_{10} J_{00}^{-1} \frac{d\mu}{d\theta} - \frac{d\mu}{d\gamma}$ is *[intuitive explanation?]*,
- $Q_S = (\pi Q^{-1} \pi^T)^{-1}$, $Q^{-1} = J_{11}$, *[intuitive explanation?]*
- J is the information (inverse covariance) matrix under the wide model divided by n, and subscripts 0 and 1 select the rows and columns forming the submatrices of J that correspond to parameters $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$ respectively,
- π is the projection matrix that selects entries of submodel S.

The estimator for the variance of $\hat{\mu}_S$ under the wide model, derived by Claeskens and Hjort (2003), is

$$\hat{V} = (\hat{\tau}^2 + \hat{\omega}^T Q_S^0 \hat{\omega})/n$$

where $\hat{\tau}^2/n$ is the variance of $\hat{\mu}_S$ under submodel S, and the additional term $(\hat{\omega}^T Q_S^0 \hat{\omega})/n$ is the increase in variance we accept by using a misspecified model S. $\hat{\tau}^2 = \frac{d\mu}{d\theta}^T J_{00}^{-1} \frac{d\mu}{d\theta}$, and $Q_S^0 = \pi^T Q_S \pi$.

Thus we compare models on the basis of the root mean square error, estimated by

$$\sqrt{\widehat{MSE}} = \sqrt{\widehat{B^2} + \hat{V}} \tag{2}$$

2.1. Bias-corrected MSE

Claeskens and Hjort (2003) derive a further correction for the bias estimator which is necessary when the above estimate is negative. The adjusted squared bias estimator is

$$\hat{B}^{*2} = max\left\{0, \quad \omega^T (I - G_S)(\hat{\boldsymbol{\delta}}\hat{\boldsymbol{\delta}}^T - Q)(I - G_S)^T \omega/n\right\} \tag{3}$$

The corresponding estimate of the bias B is $sign(\psi_{full} - \psi_S)\sqrt{\hat{B}^{*2}}$, and the bias-corrected root MSE is $\sqrt{\widehat{MSE}} = \sqrt{\hat{B}^{*2} + \hat{V}}$.

2.2. Ingredients needed for the MSE calculation

Therefore in order to calculate the MSE, we just need to know

- the estimates $\hat{\boldsymbol{\theta}}_W$ and $\hat{\boldsymbol{\gamma}}_W$ under the wide model,
- the information matrix J or covariance matrix of these estimates,
- the focus function $\mu(\boldsymbol{\theta}, \boldsymbol{\gamma})$ and its derivatives, evaluated at $\hat{\boldsymbol{\theta}}_W, \hat{\boldsymbol{\gamma}}_W$,
- the definition of which parameters are included in submodel S.

Claeskens and Hjort (2003) define the "focused information criterion" (FIC), which has a slightly simpler form due to excluding terms common to all submodels S, and is related to the MSE as

$$FIC = MSE - \hat{\tau}^2 + \omega^T Q \omega \tag{4}$$

Models with lower FIC give better estimates of the focus quantity. However we prefer to use the (root) MSE as the model comparison statistic, due to its direct interpretation as the error of the focus estimate.

2.3. Average MSE over a range of focuses

Often we want a model that performs well in a range of situations. In covariate selection problems, for example, we might want to estimate a focus quantity accurately for a defined range of covariate values. We might simply define the "averaged MSE"

$$AMSE = \int MSE(u) dW(u) du$$

as a weighted average of the mean squared errors (1) for focuses defined by different covariate values u, weighted by their prevalence $W(u)$. However Claeskens and Hjort (2008) derived an alternative formula, which includes a bias correction analogous to (3) that only needs to be performed once.

$$AMSE = max(IS, 0) + IIS \tag{5}$$

where $IS = Tr((I - G_S)(\hat{\delta}\hat{\delta}^T - Q)(I - G_S)^T A)$, $IIS = Tr(Q_S^0 A)$

$$A = J_{10} J_{00}^{-1} B_{00} J_{00}^{-1} J_{01} - J_{10} J_{00}^{-1} B_{01} - B_{10} J_{00}^{-1} J_{01} + B_{11}$$

$$B = \int \begin{pmatrix} d\mu(u)/d\boldsymbol{\theta} \\ d\mu(u)/d\gamma \end{pmatrix} \begin{pmatrix} d\mu(u)/d\boldsymbol{\theta} \\ d\mu(u)/d\gamma \end{pmatrix}^T dW(u) = \begin{pmatrix} B_{00} & B_{01} \\ B_{10} & B_{11} \end{pmatrix}$$

This is equivalent to AIC [where we average over all covariates in the data ?]

[What about equivalence to AIC where the focus is the log-likelihood?]

3. Example: covariate selection in logistic regression

This example is used by Claeskens and Hjort (2008) (Example 6.1) to illustrate the focused information criterion. The dataset was originally presented by Hosmer and Lemeshow (1989). Data are taken from $n = 189$ women with newborn babies, and the binary outcome is whether the baby is born with a weight less than 2500g. We build a logistic regression model to predict the outcome, but are uncertain about what covariates should be included.

The data are provided as an object `birthwt` in the `fic` package. This is the same as `birthwt` in `MASS` (Venables and Ripley 2002) with the addition of a few extra columns defining interactions and transformations as in Claeskens and Hjort (2008).

The following covariates are always included (coefficient vector $\boldsymbol{\theta}$)

- x_1 Weight of mother in kg, `lwtkg`

The following covariates will be selected from (coefficient vector γ)

- z_1 age, in years, `age`
- z_2 indicator for smoking, `smoke`
- z_3 history of hypertension, `ht`
- z_4 uterine irritability, `ui`
- interaction $z_5 = z_1 z_2$ between smoking and age, `smokeage`
- interaction $z_6 = z_2 z_4$ between smoking and uterine irritability, `smokeui`

Firstly the wide model, that includes all the above covariates, is defined and fitted.

```
library(fic)
wide.glm <- glm(low ~ lwtkg + age + smoke + ht + ui + smokeage + smokeui,
                data=birthwt, family=binomial)
```

The *focus function* is then defined. This should be an R function, mapping the parameters `par` of the wide model to the quantity of interest. The focus can optionally have an second argument. If supplied, this must be called `X`, and can be used to supply covariate values at which the focus function should be evaluated. Here we take the probability of low birth weight as the focus, for two covariate categories:

1. smokers with average or typical values of the other covariates. These values are given in the order supplied when specifying the model (for smokers: intercept, `lwtkg`=58.24, `age`=22.95, `smoke`=1, `ht`=0, `ui`=0, `smokeage`=22.95, `smokeui`=0).
2. non-smokers with average values of the other covariates

```
focus <- function(par, X)plogis(X %*% par)
vals.smoke <-    c(1, 58.24, 22.95, 1, 0, 0, 22.95, 0)
vals.nonsmoke <- c(1, 59.50, 23.43, 0, 0, 0, 0, 0)
X <- rbind("Smokers"=vals.smoke, "Non-smokers"=vals.nonsmoke)
```

We can illustrate these functions by calculating the probability of low birth weight, given the parameters of the fitted wide model, for each group. This is about twice as high for smokers.

```
focus(coef(wide.glm), X=X)

##              [,1]
## Smokers     0.345
## Non-smokers 0.168
```

The fic function can then be used to calculate the mean square error of the focus for one or more given submodels. For illustration we will compare two models, both including maternal weight, one including age and smoking, but the other including age, smoking and hypertension.

```
mod1.glm <- glm(low ~ lwtkg + age + smoke, data=birthwt, family=binomial)
mod2.glm <- glm(low ~ lwtkg + age + smoke + ht, data=birthwt, family=binomial)
```

We supply the following arguments to the fic function.

- wide: the fitted wide model. All the model fit statistics are computed using the estimates and covariance matrix from this model. fic will automatically recognise that this is a GLM fitted by the glm function in R, and extract the relevant information.

- inds: indicators for which parameters are included in the submodel, that is, which elements of $(\boldsymbol{\theta}, \boldsymbol{\gamma})$ are fixed to γ_0. This should have number of rows equal to the number of submodels to be assessed, and number of columns equal to $dim(\boldsymbol{\theta}) + dim(\boldsymbol{\gamma})$, the total number of parameters in the wide model, 8 in the case of wide.glm, which includes the intercept and the coefficients of seven covariates. It contains 1s in the positions where the parameter is included in the submodel, and 0s in positions where the parameter is excluded. This should always be 1 in the positions defining the narrow model, as specified in inds0 below. If just one submodel is to be assessed, inds can also be supplied as a vector of length $dim(\boldsymbol{\theta}) + dim(\boldsymbol{\gamma})$.

  Note that inds indexes *parameters* rather than *linear model terms*, that is, in covariate selection problems where a variable is a factor with more than two levels, inds should contain separate entries for the coefficient of each factor level relative to the baseline level, not just one entry indicating the presence of the factor as a whole. *[TODO can we construct these automatically?]*

- inds0 vector of indicators for which parameters are included in the narrow model, in the same format as inds. This can be omitted, in which case the narrow model is assumed to be given by the first row of inds. In this case, just the first two parameters are included, the intercept and the coefficient of lwtkg.

```
inds <- rbind(mod1 = c(1,1,1,1,0,0,0,0),
              mod2 = c(1,1,1,1,1,0,0,0))
inds0 <- c(1,1,0,0,0,0,0,0)
```

- focus the focus function.

- sub a list of the fitted submodels to be assessed. This is optional, and is only needed so that the model comparison statistics can be presented alongside the estimates of the focus under the submodels. Note, if just one submodel is to be assessed, this should still be enclosed in a list, e.g. list(mod1.glm).

The main fic function then returns an object containing the model fit statistics and the estimate of the focus quantity for each model.

```
sub <- list(mod1.glm, mod2.glm)
fic1 <- fic(wide=wide.glm, inds=inds, inds0=inds0, focus=focus, X=X, sub=sub)
fic1

##             vals mods   FIC   rmse rmse.adj   bias bias.adj
## 1      Smokers mod1 1.187 0.0723   0.0723 0.0548   0.0459
## 4      Smokers mod2 0.783 0.0556   0.0572 0.0237   0.0000
## 2 Non-smokers mod1 1.305 0.0804   0.0804 0.0765   0.0731
## 5 Non-smokers mod2 0.755 0.0596   0.0596 0.0525   0.0484
## 3          ave mod1 1.246 0.0844   0.0764 0.0657   0.0610
## 6          ave mod2 0.769 0.0678   0.0576 0.0381   0.0329
##        se focus
## 1 0.0558 0.398
## 4 0.0572 0.366
## 2 0.0334 0.243
## 5 0.0348 0.215
## 3 0.0460 0.320
## 6 0.0473 0.291
```

The object returned by `fic` is a matrix containing one row for each combination of focus covariate values indicated in the column `vals` and submodels indicated in the column `mods`. The focus estimate is returned in the final column `focus`, while the remaining columns contain the following model comparison statistics:

- `FIC` The FIC as originally defined by **??** (equation 4),
- `rmse` The root mean square error of the submodel focus estimate, calculated assuming the wide model is true (equation 2),
- `rmse.adj` The bias-adjusted root mean square error (Section 2.1),
- `bias` The estimated bias $\sqrt{\widehat{B^2}}$ (which may be undefined if $\widehat{B^2}$ is negative),
- `bias.adj` The adjusted bias estimate (Section 2.1),
- `se` The standard error $\sqrt{\hat{V}}$ of the submodel focus estimate, calculated assuming the wide model is true.

As well as the specific covariate categories, `fic` calculates model comparison statistics which are averaged over the categories, indicated by a value of `ave` in the column `vals`. TODO EXPLAIN WEIGHTS

Recall that `mod2` contains one more covariate than `mod1`. For each of the two focuses, and the average, the unadjusted and adjusted bias estimates are lower due to the inclusion of this covariate, while the standard error `se` is higher. Given the lower `rmse` and `rmse.adj` under `mod2`, the reduction in bias is deemed to be worth the increase in uncertainty.

3.1. Comparing a wide range of models

In covariate selection problems, we may want to examine a broad range of models. The function `all_inds` (a wrapper around `expand.grid`) creates a matrix of indicators that defines all submodels spanned by a given wide model (here `wide.glm` and a narrow model (here defined by `inds0`). This function works for all classes of model objects `x` for which the `terms(x)` function is understood, which includes standard R regression models such as `lm` and `glm`. *[can we use terms.formula more generally, e.g. in flexsurvreg?]* Factors are handled naturally.

```
combs <- all_inds(wide.glm, inds0)
```

The resulting matrix can be used as the `inds` argument to `fic` to calculate focused model comparison statistics for all submodels in this example, again for a focus defined by the probability of low birth weight at covariate values defined by `X`. However before calling `fic` again, we redefine `combs` to exclude models with interactions but not both corresponding main effects.

```
combs <- with(combs,
              combs[!((smoke==0 & smokeage==1) |
                      (smoke==0 & smokeui==1) |
                      (age==0 & smokeage==1) |
                      (ui==0 & smokeui==1)),])
ficres <- fic(wide=wide.glm, inds=combs, inds0=inds0, focus=focus, X=X)
```

We can actually fit the submodels in a loop as follows, by extracting the design matrix `XZ` of the wide model and removing the first column (the intercept). At each iteration, the design matrix of the submodel is defined by extracting the columns of `XZ` indexed by the corresponding row of `combs`. The submodel is fitted by placing this submatrix `XZi` on the right hand side of the model formula passed to `glm` (with `-1` appended to instruct `glm` not to include a second intercept term, since an intercept is already included in `XZi`). The list of fitted models `sub` can then be passed to the `fic` function, so that so that the focus estimate is displayed alongside the model comparison statistics.

```
nmod <- nrow(combs)
sub <- vector(nmod, mode="list")
XZ <- model.matrix(wide.glm)
for (i in 1:nmod){
  XZi <- XZ[,which(combs[i,]==1)]
  sub[[i]] <- glm(low ~ XZi - 1, data=birthwt, family=binomial)
}
ficres <- fic(wide=wide.glm, inds=combs, inds0=inds0, focus=focus, X=X,
              sub=sub)
```

Notice that some of the `rmse` elements of `ficres` are `NaN`, since the first squared bias estimator $\widehat{B^2}$ is negative. The alternative estimate $\sqrt{\widehat{B^{*2}}}$, `rmse.adj`, can simply be used in these cases.

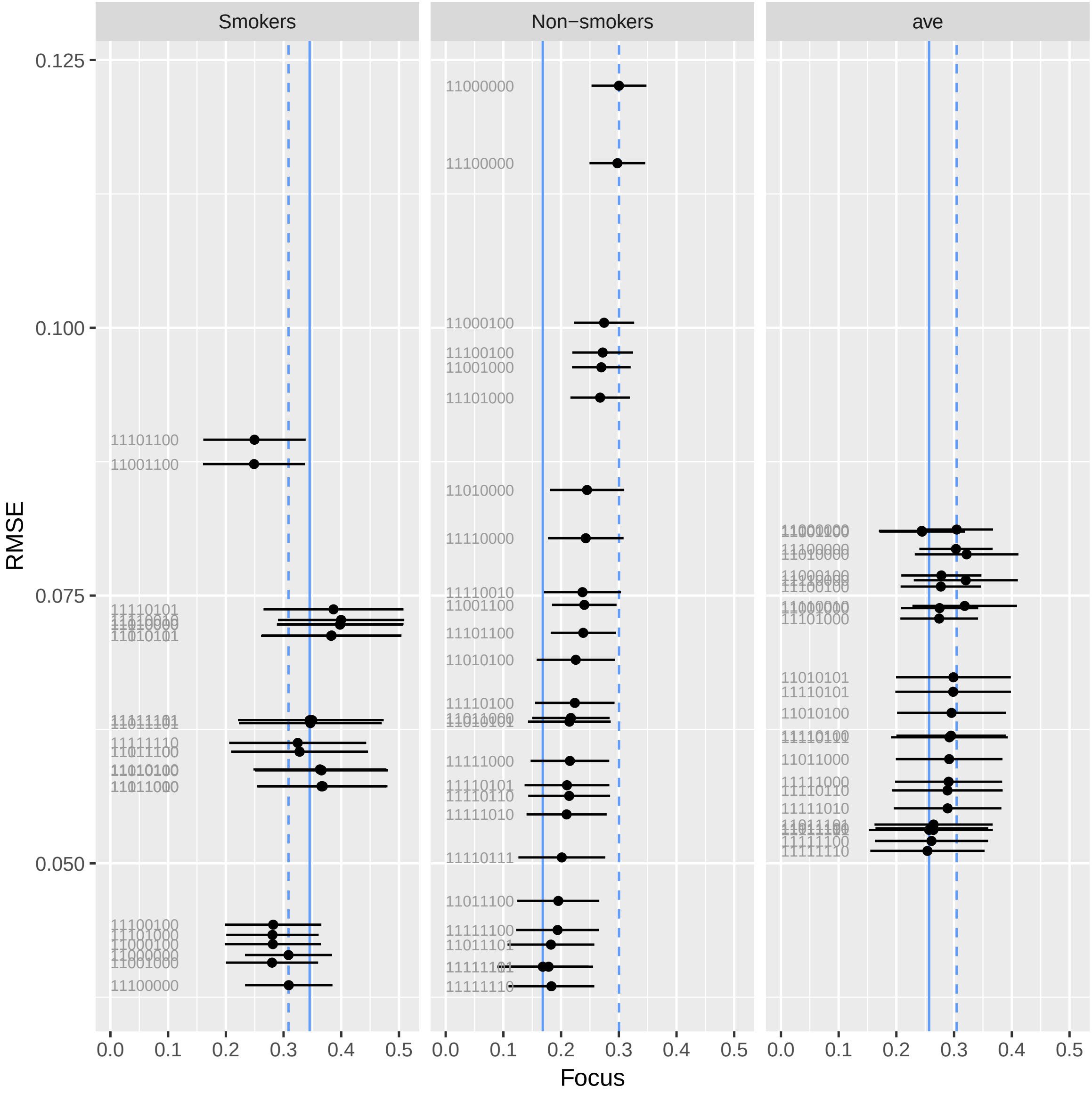
[this raises the question: why we don't just use the adjusted one all the time. Is the unadjusted one better in any circumstances?]

A comparison of many models can be illustrated by a scatterplot of the focus estimate against the RMSE of each submodel. The default `plot` method for `fic` objects accomplishes this using base R graphics: try

```r
plot(ficres)
```

Alternatively a graph can be plotted using `ggplot2` if this package is installed. This is illustrated here.

```r
ggplot_fic(ficres)
```



There is one panel for each of the two covariate categories (smokers and non-smokers) defining the focus (probability of low birth weight) and an average over the two categories. The solid blue line is the focus estimate under the wide model, and the dashed blue line is the focus estimate under the narrow model. An informal illustration of the uncertainty around the estimate of the focus quantity from each submodel is given by the estimate $\pm 1.96 \times \sqrt{\hat{V}}$. Each submodel is labelled faintly using the row names of the matrix supplied as the `inds` argument to `fic`. In this case, these names were automatically constructed by the function `all_inds` and contain a string of binary 0/1 indicators for the inclusion of eight parameters. For smokers, the narrow model (labelled `11000000`) and similar smaller models give estimates of the probability of low birth weight with the lowest MSE, while by contrast, for non-smokers, the wide model (labelled `11111111`) and similar larger models give the most accurate estimates of the focus quantity. Note that in this dataset, there are 115 non-smokers and 74 smokers, thus more data enables bigger models to be identified for non-smokers. *[discuss average?]*

4. Other classes of models

Illustrate new concepts. Could be in separate vignettes.

Linear Illustrates alternative focuses: expected outcome at given covariate values, quantile for given covariate values. Polynomial order selection as well as covariate selection. Use a well known dataset, e.g. mtcars?

Multi-state Illustrates focuses that are complicated functions of the model parameters: package should facilitate this

Survival Increasingly flexible parametric models. Challenge to express models as nested within each other. Important health economic application: restricted mean survival.

Skew-normal Novel class of models, user-written model fitting function. Store est, vcov, nobs in result list. Narrow model parameters could be in the middle.

4.1. Calling “fic” for an unfamiliar class of models

Above, the `fic` function recognised the fitted model objects as GLMs, that is, objects of class `"glm"` returned by the `glm()` function in base R. But the package can be used to calculate focused model comparison statistics for any class of models, not just the special classes it recognises. To do this, it needs to know where three things are stored inside the fitted model objects:

1. `coef`: the vector of maximum likelihood estimates $(\hat{\boldsymbol{\theta}}, \hat{\gamma})$,

2. `nobs`: the number of observations n contributing to the model fit,

3. `vcov`: the covariance matrix of the maximum likelihood estimates, $(nJ)^{-1}$.

Given a fitted model object called `mod`, the `fic()` function assumes by default that `coef(mod)`, `nobs(mod)` and `vcov(mod)` respectively return these pieces of information. If one or more of these assumptions is not true, the defaults can be changed by supplying the argument `fns` to `fic()`, which should be a named list of three components. Each component should be a function with one argument (the fitted model) which extracts the required information from the fitted model and returns it. For example, the first component of the list below is a function which, when applied to a `glm` object, returns the maximum likelihood estimates of the regression coefficients. (TODO better explained with a more obscure class?)

```
fns <- list(coef = function(x)coef(x),
            nobs = function(x)nobs(x),
            vcov = function(x)vcov(x))
fic1 <- fic(wide=wide.glm, inds=inds, inds0=inds0, focus=focus, fns=fns,
            X=X, sub=sub)
```

5. Bootstrap

Would like to

- illustrate alternative losses to the mean square error
- use resampling to illustrate FIC principles

6. Discussion

Post-selection inference, model averaging

High dimensional regression

References

Akaike H (1973). "Information theory and an extension of the maximum likelihood principle." In B Petrov, F Csaki (eds.), *2nd International Symposium on Information Theory*, pp. 267–281. Akademiai Kiado, Budapest.

Burnham KP, Anderson DR (2003). *Model selection and multimodel inference: a practical information-theoretic approach*. Springer.

Claeskens G, Hjort N (2003). "The focused information criterion (with discussion)." *Journal of the American Statistical Association*, **98**(464), 900–945.

Claeskens G, Hjort N (2008). *Model selection and model averaging*. Cambridge University Press.

Hosmer DW, Lemeshow S (1989). *Applied Logistic Regression*. John Wiley & Sons.

Kass RE, Wasserman L (1995). "A reference Bayesian test for nested hypotheses with large samples." *Journal of the American Statistical Association*, **90**, 928–934.

Schwarz G (1978). "Estimating the dimension of a model." *The Annals of Statistics*, **6**(2), 461–464.

Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. 4th edition edition. Springer.