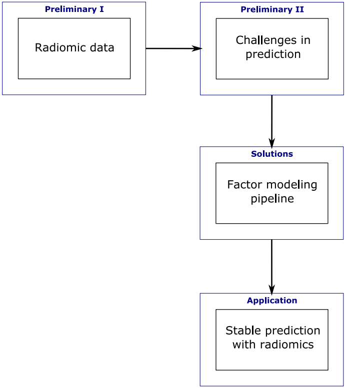
FMradio
Factor Modeling for Radiomic Data

Carel F.W. Peeters
Dept. of Epidemiology & Biostatistics
Amsterdam University medical centers, Location VUmc
cf.peeters@amsterdamumc.nl

Statistics for Omics course
Amsterdam, the Netherlands
June 28, 2019

Overview



Materials

- These slides: Contain explanation theory/methodology and exercises
- `FMradio_Practical.HTML`: Contains solutions to exercises with additional explanations
- `RadiomicsExercises.R`: Contains bare solutions to exercises

Radiomics

A recent -omics

The mining of quantitative features from medical images

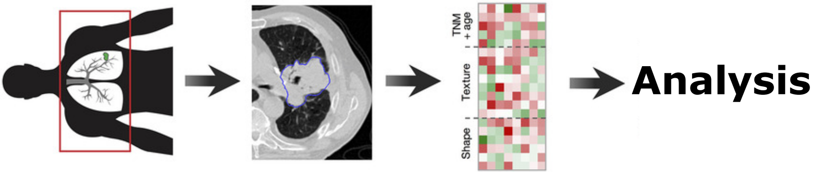
Modality

Any imaging technology, such as: computed tomography (CT), positron emission tomography (PET), and magnetic resonance imaging (MRI)

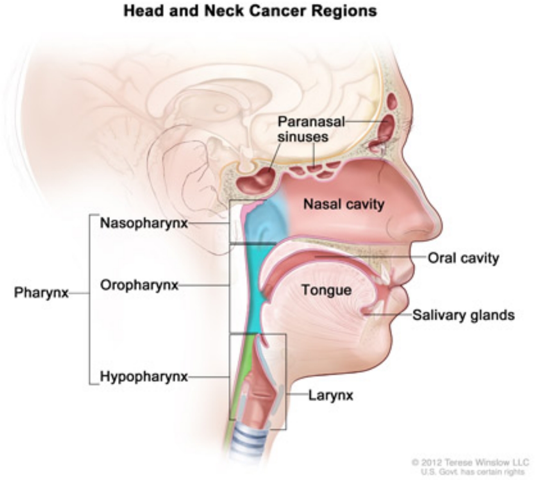
Purpose

- Expand standard radiological lexicon
- Decision support with standard-of-care images

Radiomics workflow



Head and neck cancer



Radiomic Data

Head and neck squamous cell carcinoma data

- $n = 174$ patients
- $p = 432$ radiomic features from PET/CT images

Variables (features)

Observations		1	2	3	4	5	⋯	p
	1	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}	⋯	y_{1p}
	2	y_{21}	y_{22}	y_{23}	y_{24}	y_{25}	⋯	y_{2p}
	3	y_{31}	y_{32}	y_{33}	y_{34}	y_{35}	⋯	y_{3p}
	⋮	⋮	⋮	⋮	⋮	⋮		⋮
	n	y_{n1}	y_{n2}	y_{n3}	y_{n4}	y_{n5}	⋯	y_{np}

Radiomic features

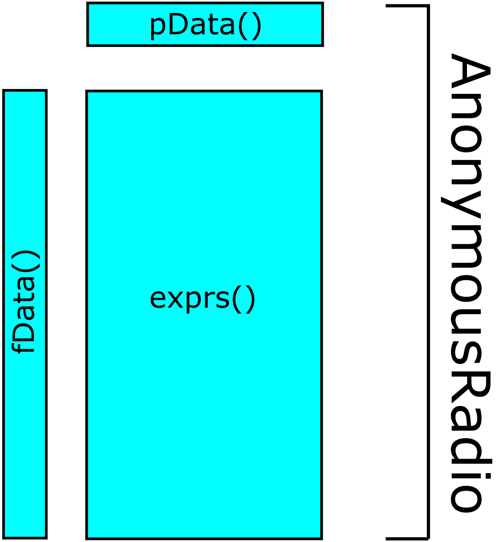
Probe feature domains such as:

- summary statistics
- morphology (shape and structure)
- texture (spatial arrangement and autocorrelation)

Prediction

Predicting survival time with event of interest being ‘death’

Radiomic Data Objects



Exercise 1: Get acquainted with the data

```
## Set working directory
setwd("")

## Needed library
library(FMradio)

## Other requirements
require("Biobase")
require("rags2ridges")
require("DandEFA")
require("randomForestSRC")
require("pec")
require("survival")
source("Convenience.R")
```

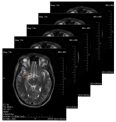
Invoke data

```
load()
```

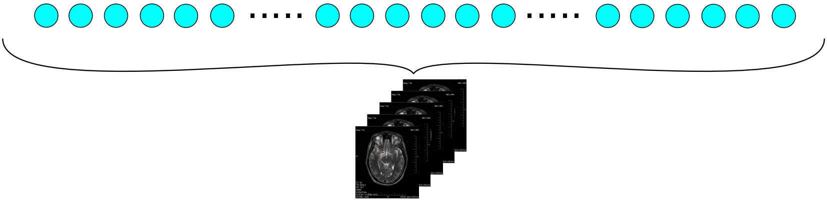
Simple exploration objects

```
head()
```

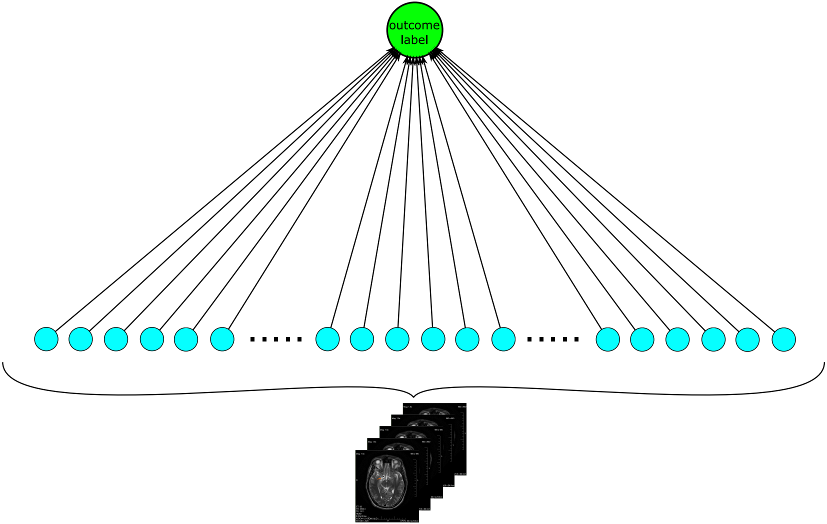
Prediction and classification



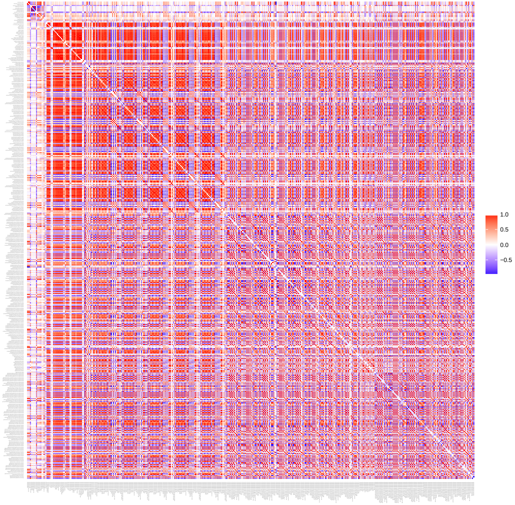
Prediction and classification



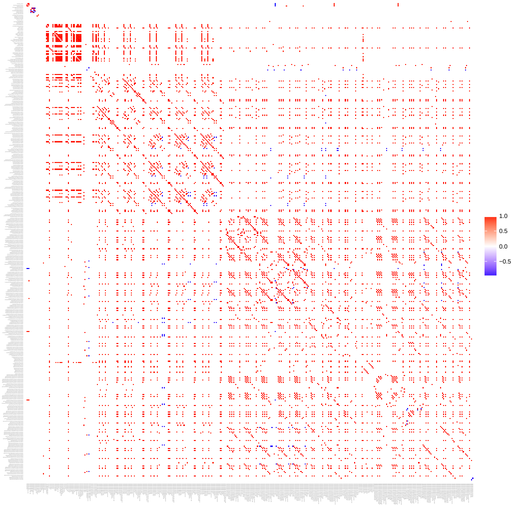
Prediction and classification



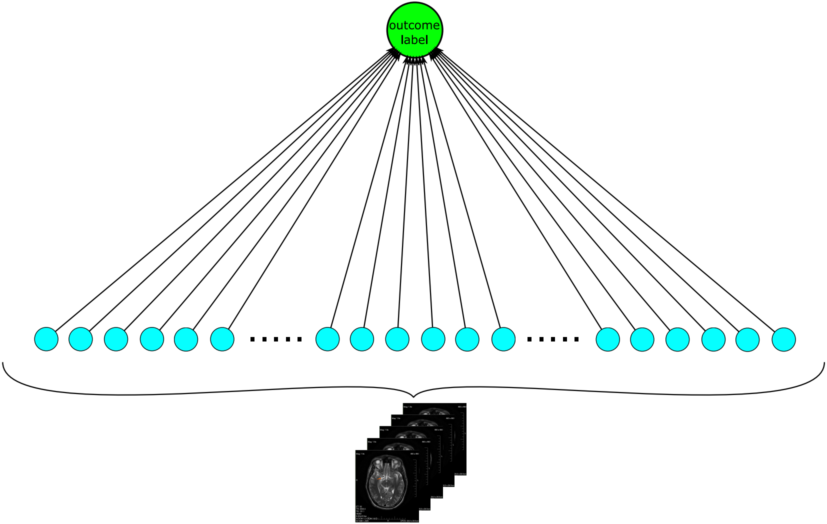
Challenge: dimension & collinearity



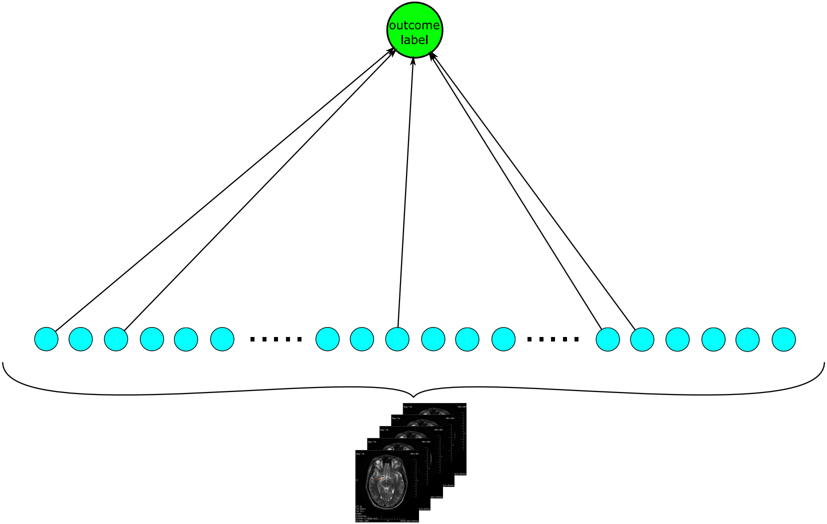
Challenge: dimension & collinearity



Challenge in prediction and classification



Challenge in prediction and classification

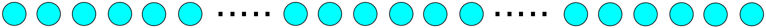


Exercise 2: Assess redundancy in the correlation matrix

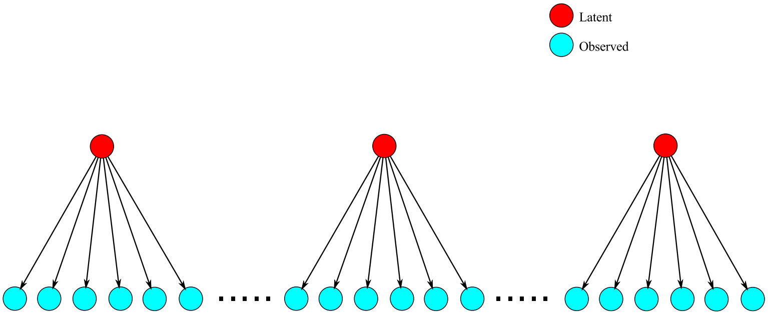
```
radioHeat(R,          ⟵ correlation matrix
          diag,       ⟵ logical
          threshold,  ⟵ logical
          threshvalue)⟵ value for thresholding
```

FMradio

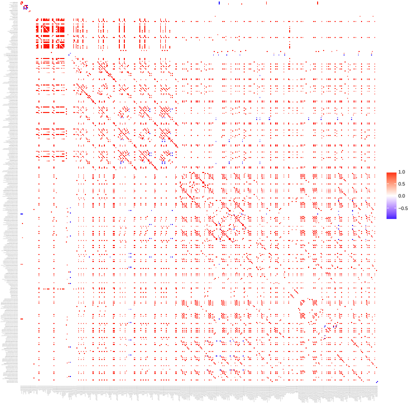
Factor analysis



Factor analysis



Redundancy filtering



Redundancy

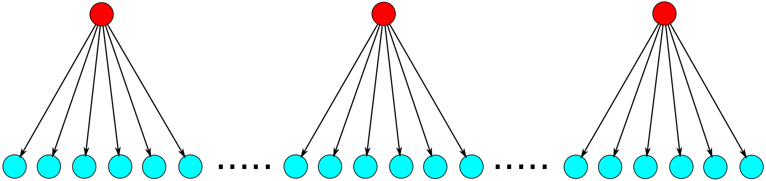
Use filtering algorithm to remove redundant features

Exercise 3: Redundancy-filter the correlation matrix

```
RF(R,        ⟵ correlation matrix
    t)       ⟵ thresholding value

subSet(X,    ⟵ data matrix or expressionset
    Rf)      ⟵ redundancy-filtered correlation matrix
```

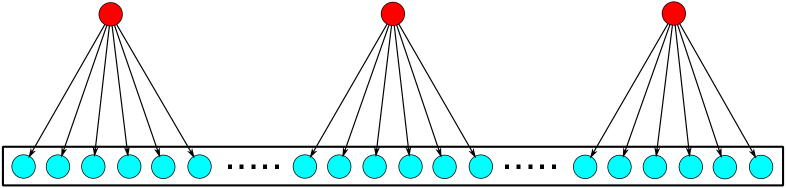
Regularized correlation matrix



Regularized estimation

$$\dot{\mathbf{R}}(\vartheta) = (1 - \vartheta)\dot{\mathbf{R}} + \vartheta \mathbf{I}_p$$

Regularized correlation matrix



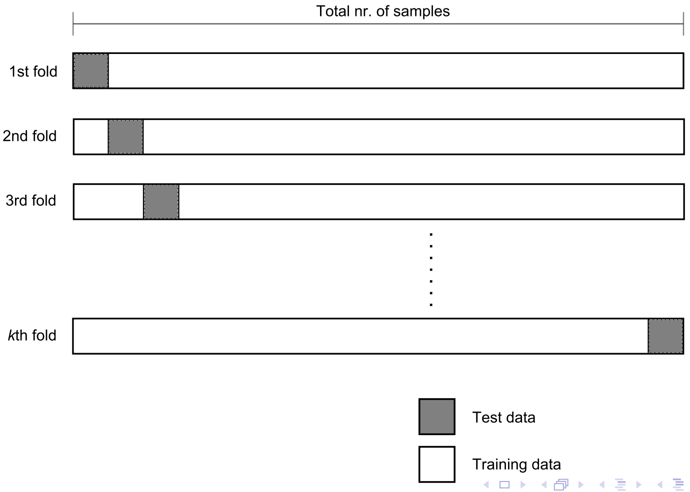
Regularized estimation

$$\dot{\mathbf{R}}(\vartheta) = (1 - \vartheta)\dot{\mathbf{R}} + \vartheta \mathbf{I}_p$$

Choosing the penalty value

K-fold cross-validation (CV)

Single iteration of *K*-fold CV



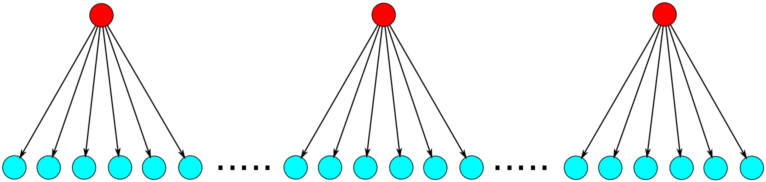
Exercise 4: Find an optimal regularized correlation matrix

```
regcor(R,     ← data matrix
       fold)  ← numeric indicating # folds
```

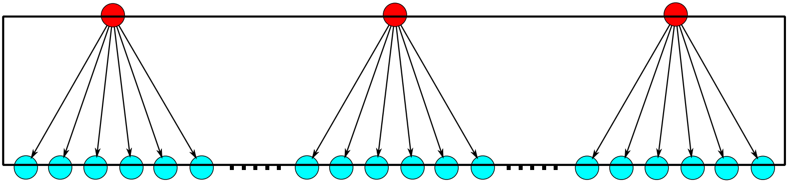
Returns list object

- `$optPen`: Optimal penalty parameter
- `$optCor`: Correlation estimate under optimal penalty parameter

Factor structure



Factor structure



Exercise 5: Perform factor analysis on the regularized correlation matrix

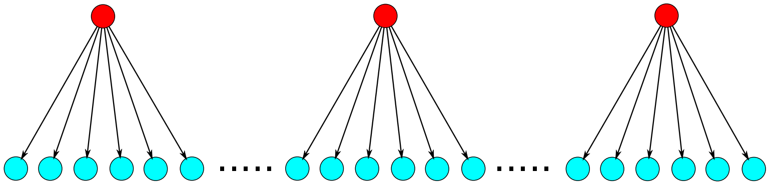
```
dimGB(R,        <——  (regularized) correlation matrix
      graph)    <——  logical

mlFA(R,         <——  (regularized) correlation matrix
     m)         <——  numeric indicating # latents
```

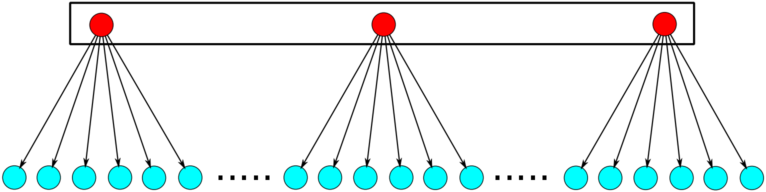
Returns

- `dimGB` can be used to support the choice of argument `m` in `mlFA`
- `mlFA` returns list object, amongst which `$Loadings`

Factor scores



Factor scores



Exercise 6: Obtain factor scores

```
facScore(X,      <——  (subsetted) data matrix
         LM,     <——  Loadings matrix
         UM,     <——  uniquenesses matrix
         type)   <——  character indicating type of score
```

Returns

Object of class `data.frame` containing the scores

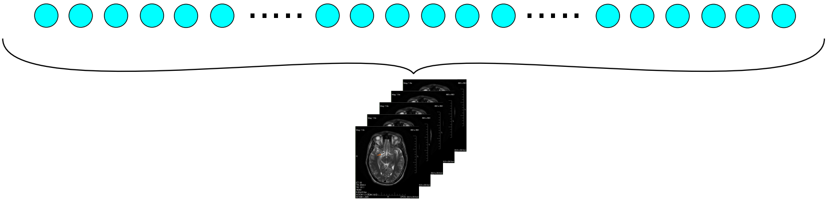
Exercise 7: Assess factor scores

```
facSMC(R,      ← (regularized) correlation matrix
       LM)     ← Loadings matrix
```

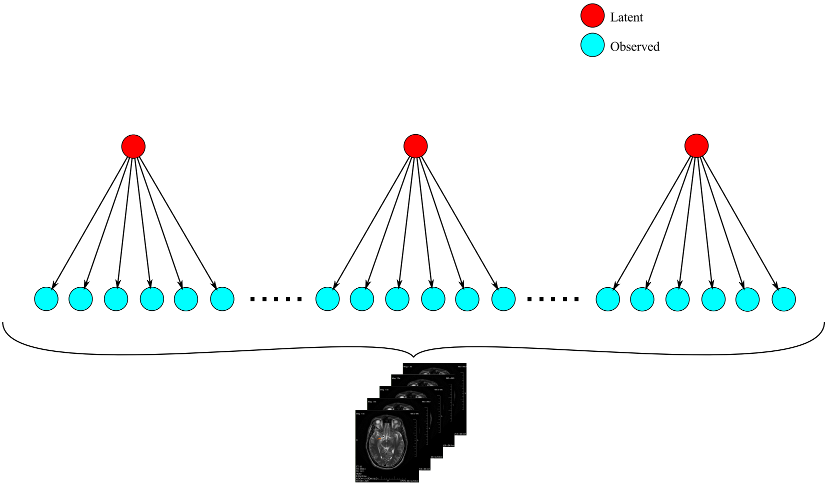
Returns

Numeric vector indicating, for each factor, the squared multiple correlation between the observed features and the latent factor

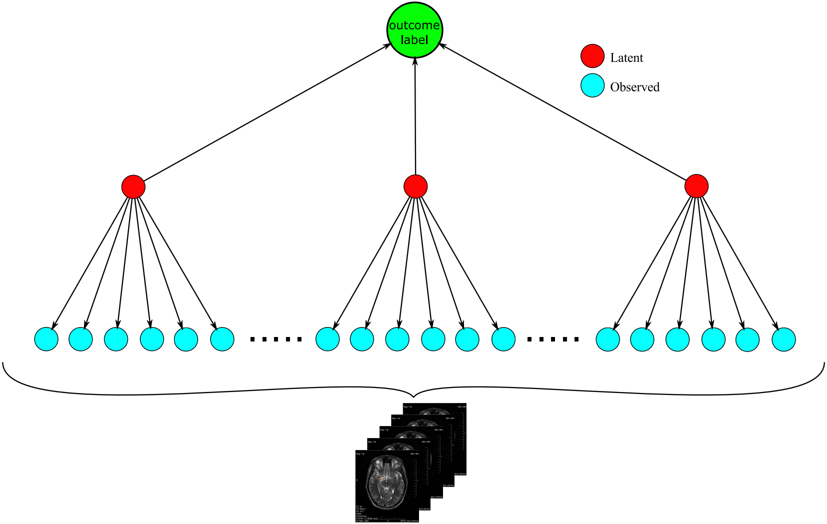
Prediction with factor scores



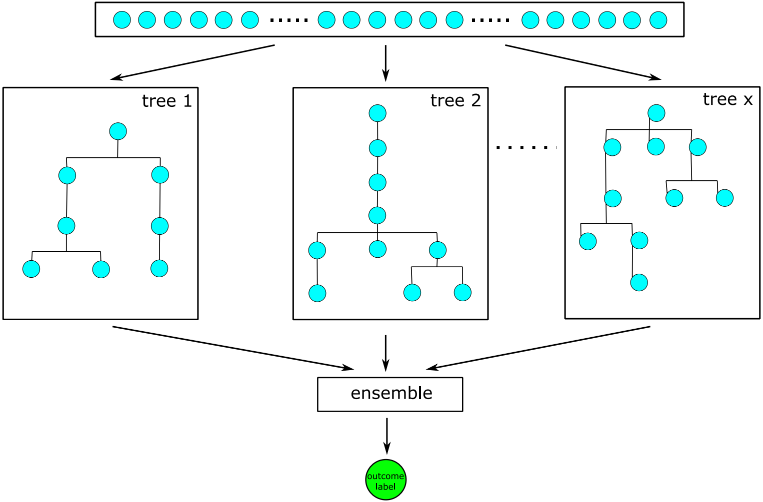
Prediction with factor scores



Prediction with factor scores



Comparison: Random Forest



Exercise 8: Concatenate original and projected data

Tip

Use `cbind()`

Exercise 8: Concatenate original and projected data

Tip

Use cbind()

```
## Combine original (scaled) data with projected meta-features
DAT <- cbind(DATAscaled, Scores)

## Include the survival information
Status  <- as.numeric(AnonymousRadio$Death_yesno) - 1
time    <- AnonymousRadio$Death_followuptime_months
DAT     <- cbind(time, Status, DAT)
```

Exercise 9: Set up model comparisons

Tip

Use `as.formula()`, `paste()`, and `list()`

Exercise 9: Set up model comparisons

Tip

Use `as.formula()`, `paste()`, and `list()`

```
## Formulating the model formula's
FitRSF     <- as.formula(paste("Surv(time, Status)~",
                               paste(colnames(DAT)[3:434],
                               collapse="+")))
FitMetaCox <- as.formula(paste("Surv(time, Status) ~",
                               paste(colnames(DAT[,c(435:442)]),
                               collapse="+")))

models <- list("MetaCox" = coxph(FitMetaCox, data = DAT,
               x = TRUE, y = TRUE),
               "RforestCox" = rfsrc(FitRSF, data = DAT))
```

Evaluation metrics

Prediction error

- Time-dependent Brier score
- Can be seen as mean square error of prediction

R^2

- Measure of (overall) explained residual variation
- Can be calculated from the Brier score

Apparent and calibrated performance

- Apparent error: prediction error obtained when the data used from training are re-used for validation
- Calibrated error: cross-validated prediction error

Exercise 10: Compare models w.r.t. prediction error

`pec(object,`	⟵	list of prediction models
`formula,`	⟵	censoring model
`data,`	⟵	data frame
`exact,`	⟵	logical determining time-points pe curve estimation
`maxtime,`	⟵	max time for estimating pe curve
`cens.model,`	⟵	method estimation inv. prob. censoring weights
`splitMethod,`	⟵	calibration method
`B)`	⟵	# of repeats/cycles

Returns

- Returns an object of class `pec`
- Can be queried using the `crps` and `Or2` functions

Results

Table: Integrated apparent and averaged cross-validated Brier scores and explained residual variations.

	$\mathcal{B}^I$		R^2	
	Apparent	Cross-validated	Apparent	Cross-validated
Reference model	.128	.130	–	–
`FMradio`	.098	**.108**	.236	**.170**
Random survival forest	.061	.114	.526	.119

Exercise 11: Visualize the results

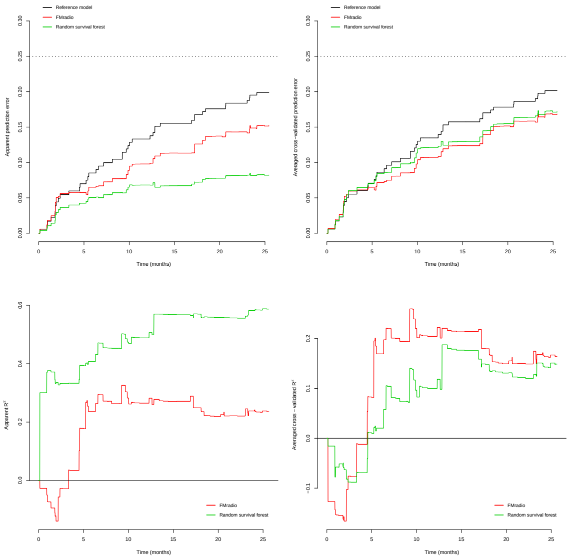
Tip

Prediction error curves can be plotted by calling the standard `plot` function on the `pec` object

Tip

R^2 curves can be plotted by using the `R2` and `plotR2Table` functions from the source-file

Visualizations



- Brent, R.P. (1971). "An Algorithm with Guaranteed Convergence for Finding a Zero of a Function". Computer Journal, 14: 422-425.
- Brier, G.W. (1950). "Verification of forecasts expressed in terms of probability". Monthly Weather Review, 78:1–3.
- Gerds, T.A. (2017). "`pec`: Prediction error curves for risk prediction models in survival analysis." R package version 2.5.4.
- Ishawaran, H., & Kogalur, U.B. (2017). `randomForestSRC`: "Random forests for survival, regression, and classification". R package version 2.5.1.
- Ishawaran, H., & Kogalur, U.B., Blackstone, E.H., & Lauer, M.S. (2008). "Random survival forests". Annals of Applied Statistics, 2(3):841–860.
- Manukyan, A., Cene, E, Sedef, A., & Demir, I. (2014). "Dandelion plot: a method for the visualization of R-mode exploratory factor analyses". Computational Statistics, 29:1769–1791.
- R. M. Martens, T. Koopman, D. P. Noij, E. Pfaehler, C. Übelhör, S. Sharma, M. R. Vergeer, C. R. Leemans, O. S. Hoekstra, Y. Maqsood, B. Zwezerijnen, M. W. Heymans, C. F. W. Peeters, R. de Bree, P. de Graaf, J. A. Castelijns, & R. Boellaard (2019). "Predictive value of quantitative ^{18}F-FDG-PET radiomics in patients with head and neck squamous cell carcinoma." Technical report, Amsterdam University Medical Centers.

Manual/Download

- Peeters, C.F.W., & Übelhör, W.N. (2019). "FMradio: Factor Modeling for Radiomics Data". R package, version 1.1. URL: https://cran.r-project.org/package=FMradio.

Theory/Methodology

- Peeters, C.F.W., Übelhör, C., Mes, S.W., Martens, R., Koopman, T., de Graaf, P., van Velden, F.P.H., Boellaard, R., Castelijns, J.A., te Beest, D.E., Heymans, M.W., & van de Wiel, M.A. (2019). "Stable prediction with radiomics data". arXiv:1903.11696.

Model

$$\mathbf{z}_i = \boldsymbol{\Lambda}\boldsymbol{\xi}_i + \boldsymbol{\epsilon}_i,$$

where

- $\boldsymbol{\Lambda}$ a $(p \times m)$-dimensional matrix of factor loadings;
- $\boldsymbol{\epsilon}_i$ denotes the error measurements for person i;
- $\boldsymbol{\xi}_i$ represent realizations of a latent variable of dimension m;
- $m < p$.

Assumptions

1. $\mathbf{z}_i \perp\!\!\!\perp \mathbf{z}_{i'}, \forall i \neq i'$;
2. $\text{rank}(\boldsymbol{\Lambda}) = m$;
3. $\boldsymbol{\epsilon}_i \sim \mathcal{N}_p(\mathbf{0}, \boldsymbol{\Psi})$, with $\boldsymbol{\Psi} \equiv \text{diag}[\psi_{11}, \ldots, \psi_{pp}]$, and $\psi_{jj} > 0, \forall j$;
4. $\boldsymbol{\xi}_i \sim \mathcal{N}_m(\mathbf{0}, \mathbf{I}_m)$;
5. $\boldsymbol{\xi}_i \perp\!\!\!\perp \boldsymbol{\epsilon}_{i'}, \forall i, i'$.

Consequence

The likelihood of the observed data can be obtained by marginalizing over $\boldsymbol{\xi}_i$:

$$
\begin{aligned}
L(\boldsymbol{\Lambda}, \boldsymbol{\Psi}; \mathbf{Z}) &= \prod_{i=1}^{n} \int f(\mathbf{z}_i | \boldsymbol{\Lambda}, \boldsymbol{\xi}_i, \boldsymbol{\Psi}) g(\boldsymbol{\xi}_i | \mathbf{I}_m)\, \partial \boldsymbol{\xi}_i \\
&= (2\pi)^{-\frac{np}{2}} \left| \boldsymbol{\Lambda}\boldsymbol{\Lambda}^{\mathrm{T}} + \boldsymbol{\Psi} \right|^{-\frac{n}{2}} \exp\left\{ -\frac{n}{2} \operatorname{tr} \left[\mathbf{R} \left(\boldsymbol{\Lambda}\boldsymbol{\Lambda}^{\mathrm{T}} + \boldsymbol{\Psi} \right)^{-1} \right] \right\},
\end{aligned}
$$

giving that the factor decomposition constrains the correlation structure of the $\mathbf{z}_i$ to be a function of $\boldsymbol{\Theta} = \{\boldsymbol{\Lambda}, \boldsymbol{\Psi}\}$, i.e.:

$$
\boldsymbol{\Sigma}(\boldsymbol{\Theta}) = \boldsymbol{\Lambda}\boldsymbol{\Lambda}^{\mathrm{T}} + \boldsymbol{\Psi}.
$$

Discrepancy function

$$F[\mathbf{\Sigma}(\mathbf{\Theta});\mathbf{R}] = \ln|\mathbf{\Sigma}(\mathbf{\Theta})| + \mathrm{tr}\left[\mathbf{R}\mathbf{\Sigma}(\mathbf{\Theta})^{-1}\right] - \ln|\mathbf{R}| - p.$$

Minimizing argument

$$\hat{\mathbf{\Theta}} := \underset{\mathbf{\Lambda},\mathbf{\Psi}}{\arg\min}\, F\left[\mathbf{\Sigma}(\mathbf{\Theta});\mathbf{R}\right],$$

such that $\hat{\mathbf{\Lambda}}^{\mathrm{T}}\hat{\mathbf{\Psi}}^{-1}\hat{\mathbf{\Lambda}}$ is diagonal and ordered.

Algorithm 1 (Redundancy filter).

Input: $\mathbf{R} \in \mathbb{R}^{p \times p}$ ▷ Raw correlation matrix
Input: τ ▷ Thresholding value

procedure $\mathrm{RF}(\mathbf{R}, \tau)$
  Go $= \mathrm{TRUE}$ ▷ Initializing logical
  create $\boldsymbol{v}$ ▷ Empty vector
  while Go **do**
    for $j = 1$ to row-dimension $\mathbf{R}$ **do**
      $\boldsymbol{v}[j] \leftarrow \sum_{j'} \mathbb{1}_{\{|\mathbf{R}[j,j']| \geq \tau\}}$
    end for
    $c \leftarrow \mathrm{which}(\boldsymbol{v} = \max(\boldsymbol{v}))[1]$ ▷ Retain index
    if $\max(\boldsymbol{v}) < 2$ **then**
      Go $= \mathrm{FALSE}$
    else
      $\mathbf{R} \leftarrow \mathbf{R}[-c, -c]$ ▷ Removing corresponding row and column
      empty $\boldsymbol{v}$
    end if
  end while
  return $\mathbf{R} \in \mathbb{R}^{p^{\dagger} \times p^{\dagger}}$
end procedure

Penalized ML representation of correlation matrix

- $\dot{\mathbf{R}} \in \mathbb{R}^{p^\dagger \times p^\dagger}$: Correlation matrix after redundancy filtering
- $\dot{\mathbf{R}}(\vartheta) = (1-\vartheta)\dot{\mathbf{R}} + \vartheta \mathbf{I}_{p^\dagger}$: Penalized correlation matrix

Choosing a value for ϑ

The kCV score for $\dot{\mathbf{R}}(\vartheta)$ based on a fixed choice of ϑ can be stated as:

$$\varphi^K(\vartheta) := \frac{1}{K}\sum_{k=1}^{K} n_k \left\{ \ln\left|\dot{\mathbf{R}}(\vartheta)_{\neg k}\right| + \mathrm{tr}\left[\dot{\mathbf{R}}_k \left(\dot{\mathbf{R}}(\vartheta)_{\neg k}\right)^{-1}\right]\right\},$$

We then choose an optimal ϑ, denoted ϑ^* such that:

$$\vartheta^* := \underset{\vartheta \in (0,1]}{\arg\min}\, \varphi^K(\vartheta).$$

Model implication

Reduced correlation matrix is Gramian and of rank m:

$$\mathbf{\Sigma}(\mathbf{\Theta}) - \mathbf{\Psi} = \mathbf{\Lambda}\mathbf{\Lambda}^{\mathrm{T}}.$$

Sample perspective

$$\dot{\mathbf{R}}(\vartheta^*) - \mathbf{I}_{p^\dagger} = (1 - \vartheta^*)(\dot{\mathbf{R}} - \mathbf{I}_{p^\dagger}), \tag{1}$$

giving the matrix we assume Gramian and whose rank we want to determine.

Implications for eigenvalues

Note (1) can be written as

$$\mathbf{V}\Big\{(1 - \vartheta^*)\left[\mathbf{D}(\dot{\mathbf{R}}) - \mathbf{I}_{p^\dagger}\right]\Big\}\mathbf{V}^{\mathrm{T}},$$

giving that the eigenvalues of interest are of the form $(1 - \vartheta^*)[d(\dot{\mathbf{R}})_j - 1]$.

Decisional rule for choosing the dimension m

$$\tilde{m} := \mathrm{card}(\mathrm{A}), \quad \text{with } \mathrm{A} \equiv \Big\{j : (1 - \vartheta^*)[d(\dot{\mathbf{R}})_j - 1] > 0\Big\}.$$

Minimizing argument

$$\hat{\boldsymbol{\Theta}} := \arg\min_{\boldsymbol{\Lambda},\boldsymbol{\Psi}} F\left[\boldsymbol{\Sigma}(\boldsymbol{\Theta}); \dot{\mathbf{R}}(\vartheta^*), \tilde{m}\right],$$

such that $\hat{\boldsymbol{\Lambda}}^{\mathrm{T}}\hat{\boldsymbol{\Psi}}^{-1}\hat{\boldsymbol{\Lambda}}$ is diagonal and ordered.

Simple structure rotation

- Let $\boldsymbol{\Gamma}$ be the Varimax rotation matrix.
- Varimax-rotated solution: $\hat{\boldsymbol{\Lambda}}\boldsymbol{\Gamma} \equiv \hat{\boldsymbol{\Lambda}}_V$.

Factor projection

Regress $\boldsymbol{\Xi} \in \mathbb{R}^{n \times \tilde{m}}$ on $\mathbf{Z}^{\dagger} \in \mathbb{R}^{n \times p^{\dagger}}$ to obtain:

$$\hat{\boldsymbol{\Xi}} = \mathbf{Z}^{\dagger}\hat{\boldsymbol{\Psi}}^{-1}\hat{\boldsymbol{\Lambda}}_V\left(\mathbf{I}_{\tilde{m}} + \hat{\boldsymbol{\Lambda}}_V^{\mathrm{T}}\hat{\boldsymbol{\Psi}}^{-1}\hat{\boldsymbol{\Lambda}}_V\right)^{-1}.$$

- The original variable-dimension ($p^{\dagger}$) is now projected onto a lower-dimensional space ($\tilde{m}$).
- Note that these scores are orthogonal by construction.

$$g\left\{\mathbb{E}(\mathbf{y})\right\} = \hat{\boldsymbol{\Xi}}\boldsymbol{\beta}$$