Module scope

ST4060/ST6015/ST6040 aims to provide a broad understanding of methodological and implementational aspects involved with current techniques used for **statistical learning**.

This includes reviewing various statistical concepts and techniques used in data exploration and analysis, methods for simulating statistical frameworks, and basic concepts of machine learning.

The objective is not so much to cover these techniques in depth (this will be done in the follow-on course), but rather to develop a sensitivity to different aspects and issues related to statistical analysis, from a practical angle.

The course commences at a basic level, and aims for you to be **self-sufficient** at R-based development of statistical analyses.

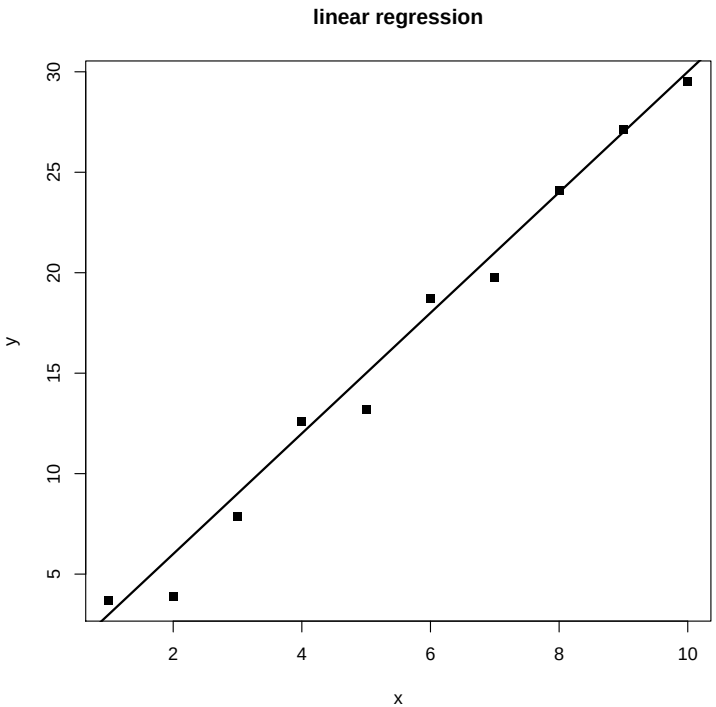
Why should you be interested?

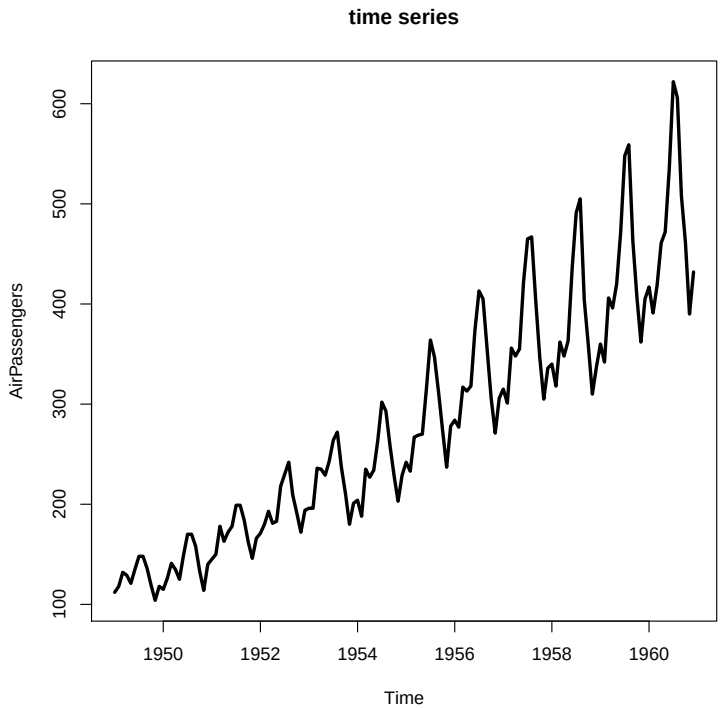
- Employers really like applied experience (quant roles, ...)
- First day at your new job: hit the ground running!

- Exploratory analysis
- Performance analysis
- Benchmarking

How is this course useful?

- Stochastic modelling
- Regression & GLM
- Survival Analysis
- Time Series Analysis
- Projects (simulation work)...
- Predicting stuff

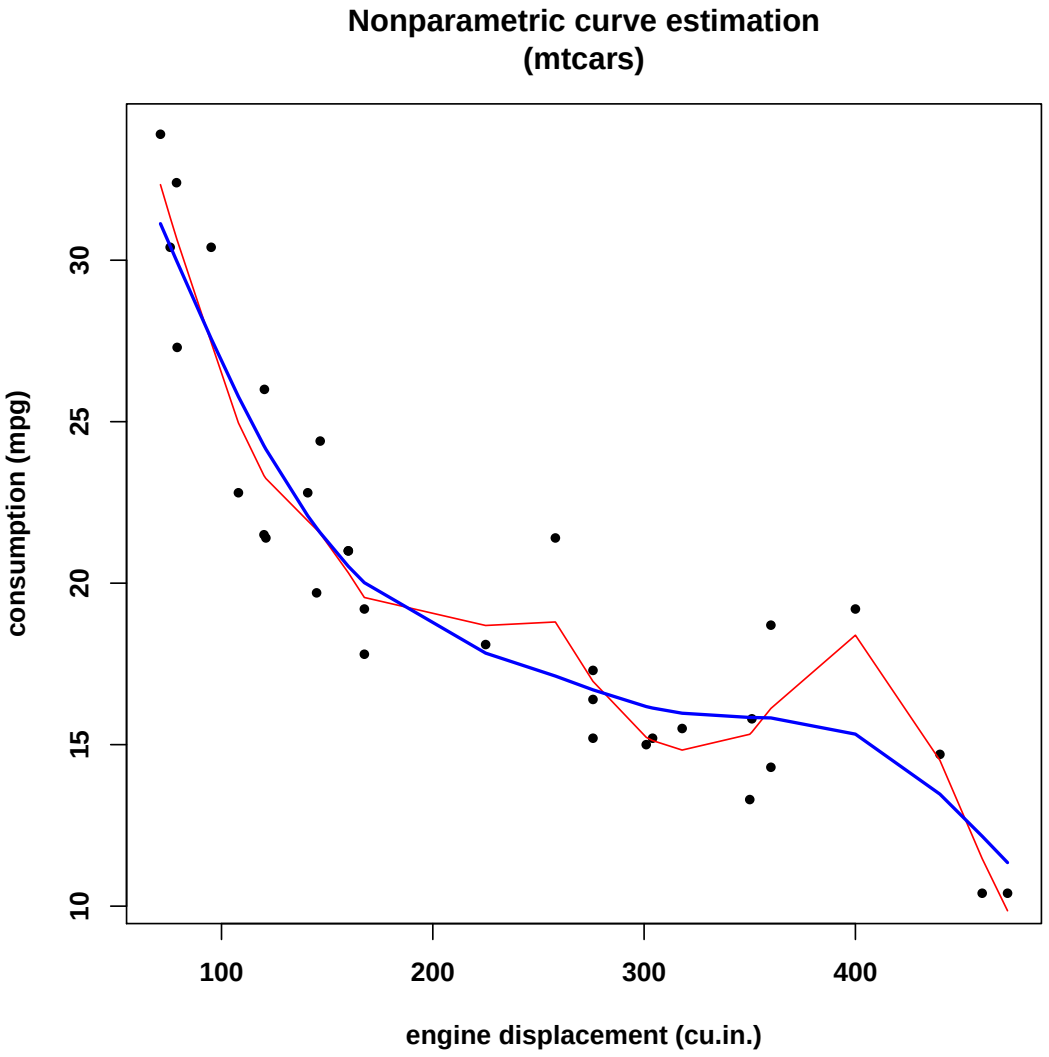




Course outline

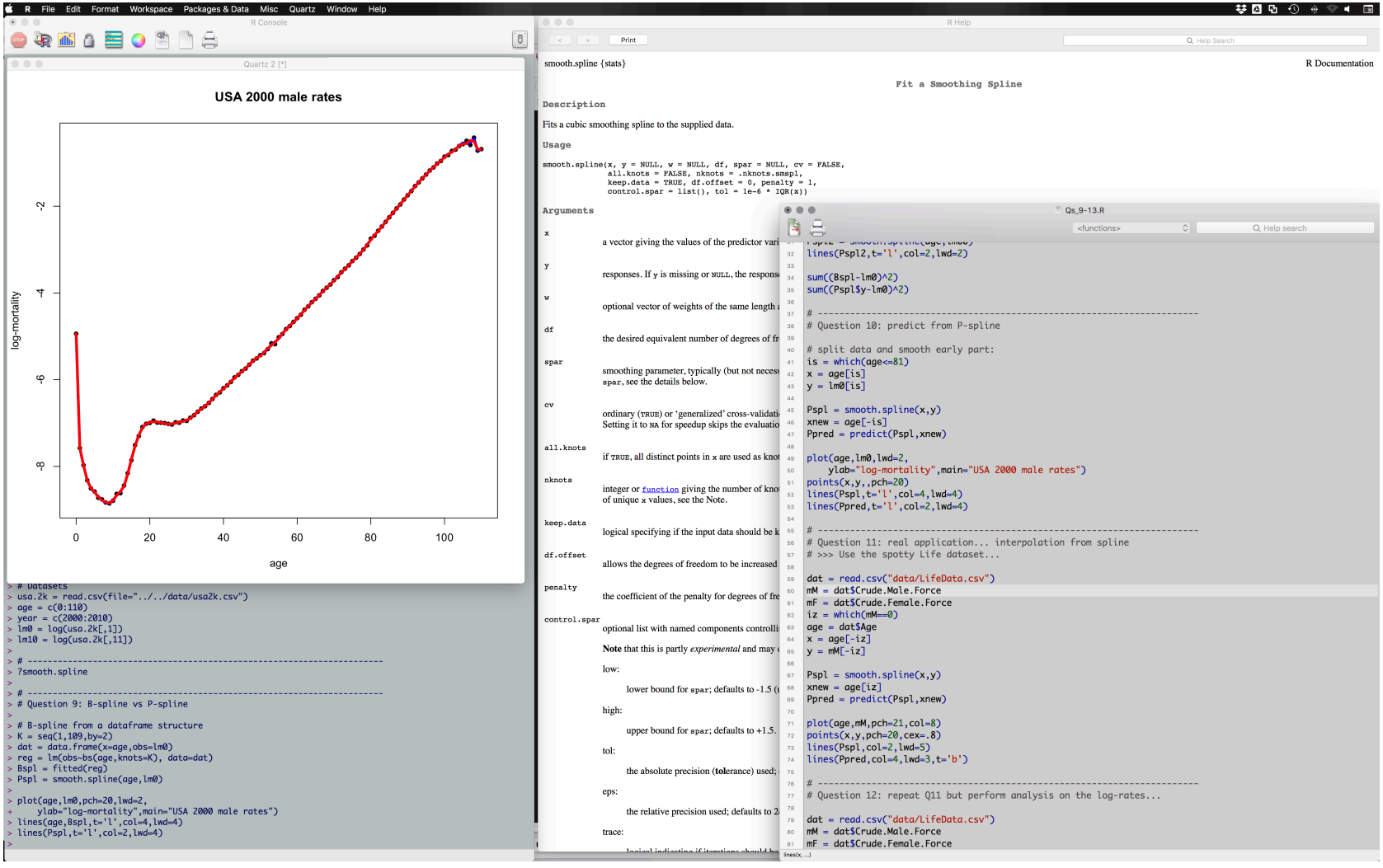
1. **Stochastic modelling**
   Why and how we model random stuff
2. **Resampling**
   How to simulate data and estimation processes
3. **Regression & Parametric models**
   A battery of tools to describe typical patterns
4. **Smoothing & nonparametric modelling**
   Other tools to describe patterns without assuming their shape
5. **Fundamentals of Statistical Learning**
   Overview of some of the main problems met in Statistical (and Machine) Learning

Example: nonparametric curve estimation

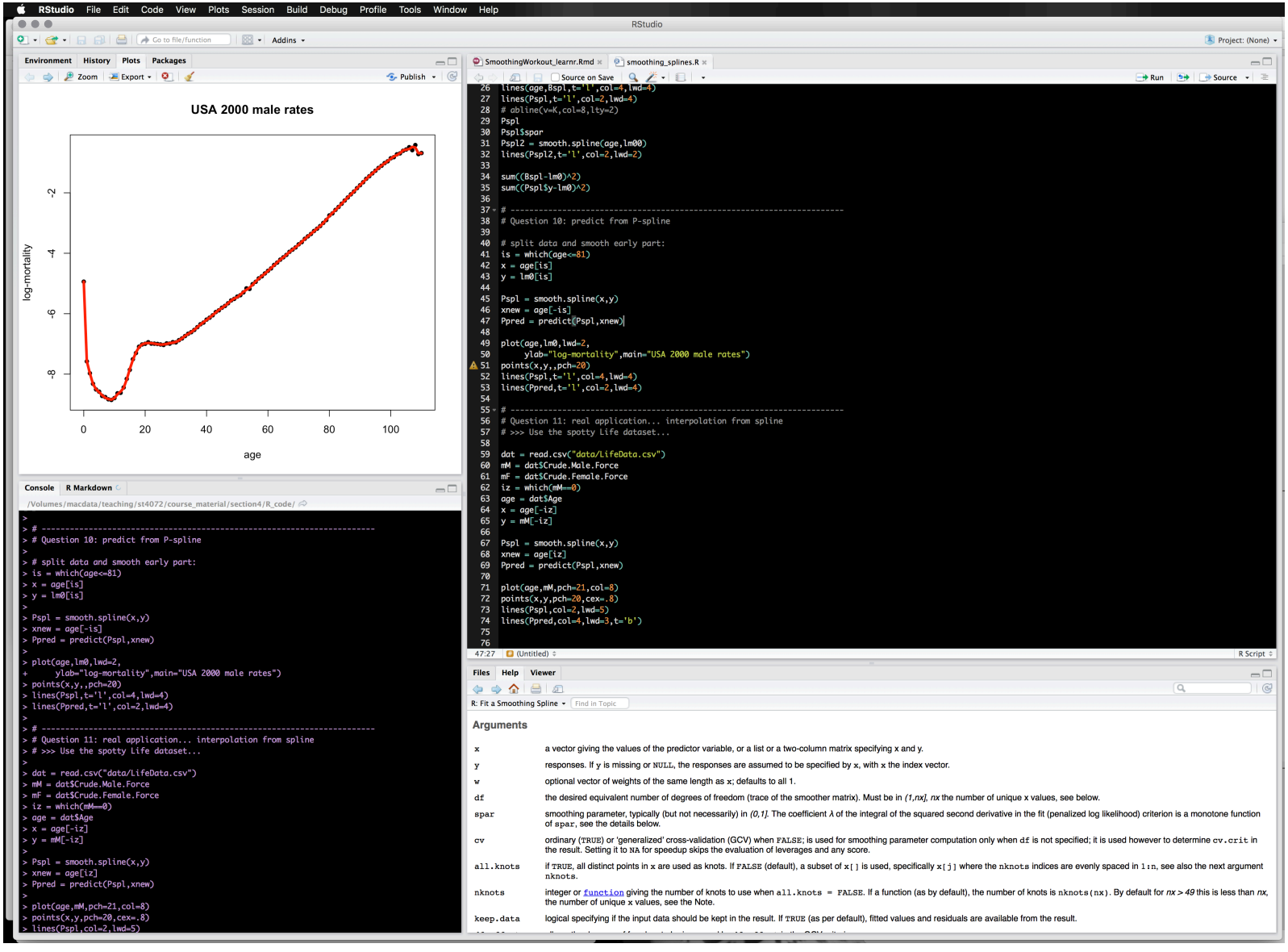


- How do we come up with these models (i.e. curves)?
- Why is the blue model better than the other one?
- How do we measure this performance?
- And then what?

Using R...


R File Edit Format Workspace Packages & Data Misc Quartz Window Help
R Console
Quartz 2 [*]
USA 2000 male rates
log-mortality
age
R Help
Print
smooth.spline {stats}
R Documentation
Fit a Smoothing Spline
Description
Fits a cubic smoothing spline to the supplied data.
Usage
smooth.spline(x, y = NULL, w = NULL, df, spar = NULL, cv = FALSE,
all.knots = FALSE, nknots = .nknots.smspl,
keep.data = TRUE, df.offset = 0, penalty = 1,
control.spar = list(), tol = 1e-6 * IQR(x))
Arguments
Qs_9-13.R
# Datasets
usa.2k = read.csv(file="../../data/usa2k.csv")
age = c(0:110)
year = c(2000:2010)
lm0 = log(usa.2k[,1])
lm10 = log(usa.2k[,11])
# Question 9: B-spline vs P-spline
# B-spline from a dataframe structure
K = seq(1,109,by=2)
dat = data.frame(x=age,obs=lm0)
reg = lm(obs~bs(age,knots=K), data=dat)
Bspl = fitted(reg)
Pspl = smooth.spline(age,lm0)
plot(age,lm0,pch=20,lwd=2,
ylab="log-mortality",main="USA 2000 male rates")
lines(age,Bspl,t='l',col=4,lwd=4)
lines(Pspl,t='l',col=2,lwd=4)
lines(Pspl2,t='l',col=2,lwd=2)
sum((Bspl-lm0)^2)
sum((Pspl$y-lm0)^2)
# Question 10: predict from P-spline
# split data and smooth early part:
is = which(age<=81)
x = age[is]
y = lm0[is]
Pspl = smooth.spline(x,y)
xnew = age[-is]
Ppred = predict(Pspl,xnew)
plot(age,lm0,lwd=2,
ylab="log-mortality",main="USA 2000 male rates")
points(x,y,,pch=20)
lines(Pspl,t='l',col=4,lwd=4)
lines(Ppred,t='l',col=2,lwd=4)
# Question 11: real application... interpolation from spline
# >>> Use the spotty life dataset...
dat = read.csv("data/LifeData.csv")
mM = dat$Crude.Male.Force
mF = dat$Crude.Female.Force
iz = which(mM==0)
age = dat$Age
x = age[-iz]
y = mM[-iz]
Pspl = smooth.spline(x,y)
xnew = age[iz]
Ppred = predict(Pspl,xnew)
plot(age,mM,pch=21,col=8)
points(x,y,pch=20,cex=.8)
lines(Pspl,col=2,lwd=5)
lines(Ppred,col=4,lwd=3,t='b')
# Question 12: repeat Q11 but perform analysis on the log-rates...
dat = read.csv("data/LifeData.csv")
mM = dat$Crude.Male.Force
mF = dat$Crude.Female.Force

... or RStudio...


RStudio File Edit Code View Plots Session Build Debug Profile Tools Window Help
RStudio
Environment History Plots Packages
Zoom Export Publish
USA 2000 male rates
log-mortality
-2
-4
-6
-8
0 20 40 60 80 100
age
Console R Markdown
/Volumes/macdata/teaching/st4072/course_material/section4/R_code/
SmoothingWorkout_learnr.Rmd smoothing_splines.R
Source on Save Run Source
lines(age,Bspl,t='l',col=4,lwd=4)
lines(Pspl,t='l',col=2,lwd=4)
# abline(v=K,col=8,lty=2)
Pspl
Pspl$spar
Pspl2 = smooth.spline(age,lm00)
lines(Pspl2,t='l',col=2,lwd=2)
sum((Bspl-lm0)^2)
sum((Pspl$y-lm0)^2)
# ---
# Question 10: predict from P-spline
# split data and smooth early part:
is = which(age<=81)
x = age[is]
y = lm0[is]
Pspl = smooth.spline(x,y)
xnew = age[-is]
Ppred = predict(Pspl,xnew)
plot(age,lm0,lwd=2,
ylab="log-mortality",main="USA 2000 male rates")
points(x,y,,pch=20)
lines(Pspl,t='l',col=4,lwd=4)
lines(Ppred,t='l',col=2,lwd=4)
# ---
# Question 11: real application... interpolation from spline
# >>> Use the spotty Life dataset...
dat = read.csv("data/LifeData.csv")
mM = dat$Crude.Male.Force
mF = dat$Crude.Female.Force
iz = which(mM==0)
age = dat$Age
x = age[-iz]
y = mM[-iz]
Pspl = smooth.spline(x,y)
xnew = age[iz]
Ppred = predict(Pspl,xnew)
plot(age,mM,pch=21,col=8)
points(x,y,pch=20,cex=.8)
lines(Pspl,col=2,lwd=5)
lines(Ppred,col=4,lwd=3,t='b')
47:27 (Untitled) R Script
Files Help Viewer
R: Fit a Smoothing Spline
Find in Topic
Arguments
x a vector giving the values of the predictor variable, or a list or a two-column matrix specifying x and y.
y responses. If y is missing or NULL, the responses are assumed to be specified by x, with x the index vector.
w optional vector of weights of the same length as x; defaults to all 1.
df the desired equivalent number of degrees of freedom (trace of the smoother matrix). Must be in (1,nx], nx the number of unique x values, see below.
spar smoothing parameter, typically (but not necessarily) in (0,1]. The coefficient λ of the integral of the squared second derivative in the fit (penalized log likelihood) criterion is a monotone function of spar, see the details below.
cv ordinary (TRUE) or 'generalized' cross-validation (GCV) when FALSE; is used for smoothing parameter computation only when df is not specified; it is used however to determine cv.crit in the result. Setting it to NA for speedup skips the evaluation of leverages and any score.
all.knots if TRUE, all distinct points in x are used as knots. If FALSE (default), a subset of x[] is used, specifically x[j] where the nknots indices are evenly spaced in 1:n, see also the next argument nknots.
nknots integer or function giving the number of knots to use when all.knots = FALSE. If a function (as by default), the number of knots is nknots(nx). By default for nx > 49 this is less than nx, the number of unique x values, see the Note.
keep.data logical specifying if the input data should be kept in the result. If TRUE (as per default), fitted values and residuals are available from the result.

Interactive LearnR pages

A number of interactive online pages have been designed to help you practice on focused R aspects for each of the course sections:

- Basic R LearnR page
- Modelling basics LearnR page
- Resampling LearnR page
- Regression LearnR page
- Smoothing LearnR page (link TBA)
- Machine Learning LearnR page (link TBA)

Comments/feedback?

For any comments or queries about this course, please contact

eric.w@ucc.ie