Secondo Torino R net meeting

Torino, 17 novembre 2011

Using R
in Operational Risk Measurement:
the Intesa Sanpaolo experience

Roberto Ugoccioni

INTESA SANPAOLO

Disclaimer

Opinions and views expressed here are of the speaker exclusively and do not necessarily represent practice or policies of Intesa Sanpaolo Spa

Outline

- Introduction: operational risk
  - some definitions and examples
  - what we use R for (O.R. measurement)
- Our experience with R (since 2005, R v2.2.0)
  - reasons to use R
  - satisfactions
  - tribulations
  - a short wish-list
- Conclusions

Intesa Sanpaolo Group

- Intesa Sanpaolo is a banking group resulting from the merger, since 2007, between Banca Intesa and Sanpaolo IMI. Intesa Sanpaolo is now among the top banking groups in the Euro zone.
- In Italy the Group offers its services in all business areas (retail, corporate and wealth management) to 10.8 million customers through a network of 5,600 branches.
- It has has a selected presence in Central Eastern Europe and Middle Eastern and North African areas with more than 1,700 branches and 8.3 million customers belonging to the Group's subsidiaries operating in retail and commercial banking in 12 countries. The International network has a presence in 29 countries to support cross-border activities of corporate customers.

Risk Management

- Guarantees measurement and control, both current and perspective, of the Group exposure to the different types of risk, in particular to market, credit, interest rate, liquidity, operational and country risks.
- Develops and maintains systems for risk measurement, management and control, in agreement with the Basel 2 regulations and with international best practices.
- Follows regulations developments and provides Regulatory Authorities with information on internal models.

Basel 2 and the role of Regulatory Authorities

- 2004: New Basel Capital Accord (a.k.a. Basel II). It is the second of the Basel Accords, which are recommendations on banking laws and regulations issued by the Basel Committee on Banking Supervision (an international body which coordinates operations of central banks and monetary authorities).
  - Minimal capital requirements, separately for
    - Credit risk
    - Operational risk
    - Market risk
  - Supervisory review
  - Market discipline (transparency)

Definition of Operational Risk

- Nothing new! Operational Risk Management is about what can go wrong in the normal course of operations. It is the very subject of Murphy's Law...
- Examples for a University:
  - Equipment gets stolen
  - Power outage (with possible damage to equipment)
  - Accidents to students, personnel
  - ...
- Basel 2 definition:
  - The risk of loss resulting from inadequate or failed internal processes, people and systems or from external events.

Operational Risk for a Financial Institution

- Again: when something goes wrong in the course of normal operations, producing a monetary impact on the Institution:
  - Money, valuables get stolen
  - Power outage (with damages)
  - Accidents to customers, personnel
  - Internal frauds
  - Errors (human or software)
  - Product mis-specification or mis-selling
  - …
- NOT Operational Risk:
  - Money is lent out to a company which cannot pay back the loan (Credit Risk)

Operational Risk can be fatal

- Examples of cases where severe operational events occurred (and in some brought down a company, with damages to shareholders and/or customers):
  - Barings Bank (1995) and Société Général (2008): internal fraud
  - Long Term Capital Management (model error, 2000): model error
  - Payment Protection Insurance (2011): product mis-selling
- These are rare events with catastrophic consequences!
- Regulatory Authorities decided it was needed to take protection against such possibilities: banks are required to hold capital enough to face rare events of that type
- Computing just how much capital has to be held is the very subject of Operational Risk measurement.

OR: Advanced Measurement Approach

- Current regulations encourage a large bank to compute a capital requirement for Operational Risk based on a statistical model, developed internally (so called Advanced Measurement Approach). According to Basel 2:
  - the capital requirement is a risk measure which is compatible with the 99.9% of the annual loss distribution (i.e. the distribution of the total loss of one year)
- About the methodology there is quite some freedom, but it has to be approved by
  - internal validation unit
  - internal audit
  - national regulatory bodies (Banca d'Italia)
- Required input to the AMA model:
  - internal loss data that covers at least 5 years history
  - external loss data (data pooling)
  - scenario analysis (perspective view)
  - business environment and control factors

OR: measurement

- Because of very diverse dynamical mechanisms at work, the approach so far has been a purely statistical one, based on actuarial methodology
- Measure:
  - Frequency distribution of OR events (number of events per year, *n*)
  - Severity distribution of OR events (given that an event occurs, what is its economic impact, *x*)
- Compute:
  - Yearly loss distribution (probability of losing less than *x* in a year)

$$S(x) = \sum_n P_n \, F^{*n}(x)$$

Statistical tools

- On the road to the internal model we looked into several different statistical toolboxes:
  - distribution fitting (maximum likelihood, robust fitting, GoF)
  - Bayesian fitting (MCMC, credibility)
  - data mining
  - GLMs
  - Extreme value Theory
  - game theory
- ...as well as exploring different algorithms and implementations
  - Monte Carlo simulations
  - High Performance computing (parallel computing)
  - FFTs and copulas

- Creating a new model for a new phenomenon requires a powerful, flexible framework
  - Model development calls for a lot of interactive use (trial and error, exploration)
  - Regulatory calculation requires a non-interactive procedure

Why R, and what for

- Why use R?
  - statistical tools available in house not suitable
  - very powerful, complete stats oriented language
  - flexible framework
  - open source
  - actively maintained
  - huge list of specialized packages
  - previous experience within model building team
- How R is used at Intesa Sanpaolo:
  - methodological research
  - application prototyping
  - production environment (but: migrating now)
  - occasional data mining / custom analyses

From experience: R packages

- Several packages exist on CRAN which are specific for financial risk management and some even cater to OR modelling
  - Most of them are recent or were not mature enough when we started developing our model
  - Need to be able to understand deeply all techniques and check all implementations (some packages still young, robustness varies)
  - Developed in house most of the functionalities required
  - No fancy statistical techniques, but we stress quite a bit the non-fancy ones
  - Easy to try out different statistical tools

From experience: the R language - lists

- a `list()` is an extremely useful object: an indexable (by position and by name) flexible container, any R object can be put in a list together with any other, including functions
- this can entice you the treat them as a replacement for OOP (which R has of its own)

```
f <- fit.distribution( datapoints, pdf, start ) {
      [....]
      estimates <- optim( start, pdf )
      f$cdf <- function(x) do.call(pdf, estimates)
      [....]
}
curve( f$cdf, from=0, to=1e8, log="x" )
```

- but beware of what you put in it – functions are saved with their own environments, which may take up a lot of space; one might end up with objects too big to fit all into memory alongside each other
- sometimes too much flexibility can lead astray...

From experience: fixes and upgrades

- reported bug in RGui editor window (corrupted memory) in august 2005
  - fix committed in october
- some of our code broke when going from version 2.5 to 2.6, due to shuffling of `approxfun()` between *base* and *stats* packages

```
f <- approxfun(x,y)
save(f, file="test_approx.Rdata")
```

- no answer on this one
- difficult to follow a biannual update cycle, as we have to ensure reproducibility of all past results and limited resources
  - still using 2.6.2

From experience: IT constraints

- A large company's IT department is often inflexible (need to guarantee high service level). For us it meant:
  - Powerful machines hard to get (costly!)
  - Constraints on OS (Windows, HPUX)
  - Constraints on software installation (fortunately R does not need admin privileges)
  - IT dept not able to handle open source software (we do our own maintenance)

From experience: parallel computing

- Need to perform many repetitive time consuming tasks in a minimal amount of time (i.e., lots of tests) resulted in the use of parallel computing early on
  - mostly “trivial parallelism”
  - no MPI-type framework available in our Windows servers
- Chose the NetWorkSpaces (NWS) framework
  - straightforward sharing of variables between processes
  - effective results need targeted programming

```
s <- standardSleigh( workerCount= <N> | “web” )
standardSleighExportNames(s, “own.data”)
do.fit <- function(thr, dist) {  ..... }
result <- eachElem( s, do.fit,
     as.character( expand.grid(thresholds, distribution) )
     )
```

From experience: other languages

- Existing tool: Fortran, C code
  - used highly optimised library (for performance)
  - easy to integrate in R via external call, when compiled as DLL
  - can thus reuse existing code (with some care)

```
h <- .Fortran("FFTDLL", hdummy=as.integer(length(h)),
      h=as.double(h), as.double(tilt), as.double(a),
      as.double(b), as.integer(freqtype), DUP=FALSE)$h
```

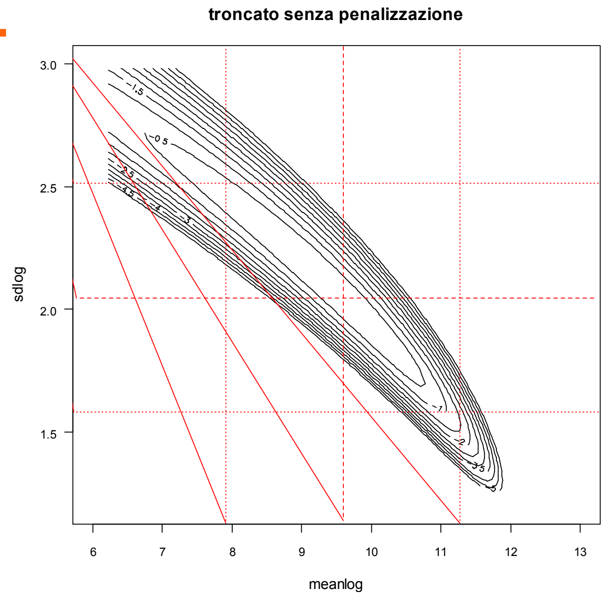
```
subroutine fftdll(nh,h, tilt, a, b, freqtype)
!DEC$ ATTRIBUTES DLLEXPORT,C,REFERENCE,ALIAS:'fftddl_' :: FFTDLL
double precision :: a, b, h, tilt
integer :: nh, i, freqtype, n
type(DFTI_DESCRIPTOR), POINTER :: My_Desc2_Handle
dimension :: h(nh)
```

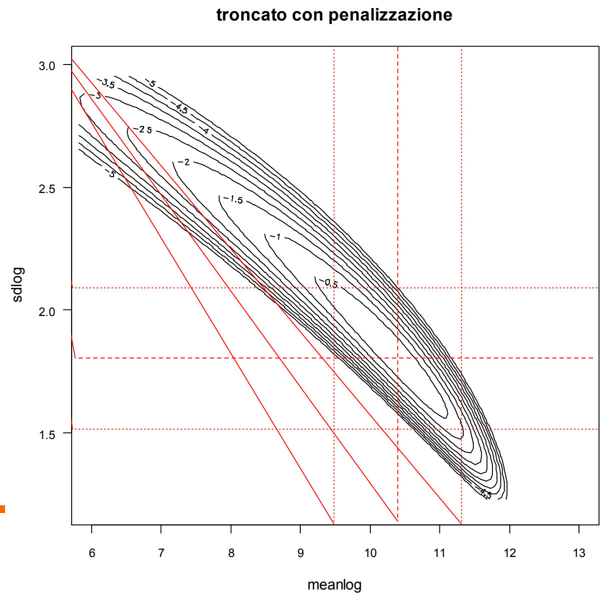
From experience: various items

- Graphs
  - large offer!
  - we use the base system
    - easier to build a plot in steps (one curve/dataset at a time), especially during development (pity legends by hand)
  - lattice is cool but cumbersome in development, except for conditioning plots
  - ggplot2 very interesting (no time to learn!)
- little things we wrote code for, in order to make life simpler:
  - vlookup()
  - data-frame build-up (with partial rows)
  - from “by” to tables / data-frame

From experience: a case study

- easy to include special cases in general framework: ML fitting of truncated distributions is error-prone
- our solutions: penalized likelihood:
  - PLL(x;θ) = LL(x;θ) – p(θ)
- the R object resulting from the fitting routine is self-contained, including functions
  - → no need to add special cases to the rest of the code





From experience: a wish-list

- interactive object browser (IDE?)
- integration with other company-standard software (Office)
  - anyway, excellent CSV reader!
- note that these issues are addressed by commercial add-ons, and open-source is getting there too (but requires recent R versions)

Conclusions

- in retrospect, happy with the decision to use R
- R is an excellent software, also in a business environment (indeed it is used by other large financial institutions)
  - easy to test/implement new ideas
  - good to have at hand for ad hoc problems