Bayesian Modelling of Loss Curves in Insurance

Mick Cooney
michael.cooney@applied.ai

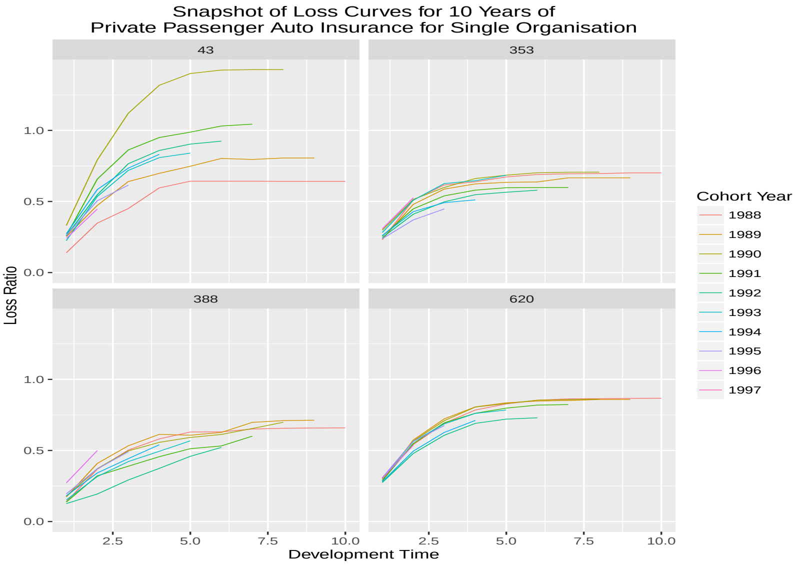
15 April 2016

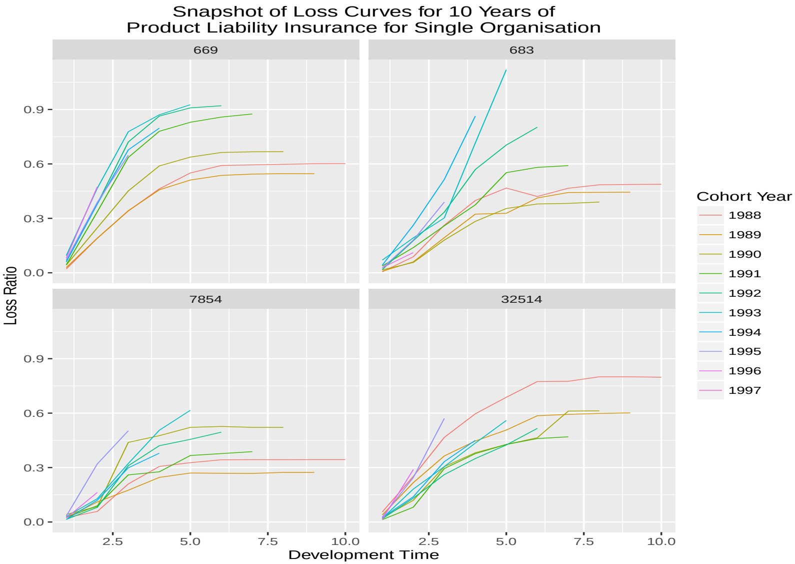
Structure of Talk

- Loss Curves
- Chain Ladder Modelling (package `ChainLadder`)
- Loss Growth Modelling
- Expanding the Model
- Posterior Predictive Checks
- Summary

Loss Curves

```
use_grcode <- c(43,353,388,620)

ppauto_ss_dt <- ppauto_dt[GRCODE %in% use_grcode
                        ][DevelopmentYear < 1998
                        ][, .(grcode      = GRCODE
                             ,accyear     = AccidentYear
                             ,devlag      = DevelopmentLag
                             ,premium     = EarnedPremDIR_B
                             ,cumloss     = CumPaidLoss_B
                             ,loss_ratio = CumPaidLoss_B / EarnedPremDIR_B)]


print(dcast(ppauto_ss_dt[grcode == 43]
           ,grcode + accyear + premium ~ devlag
           ,value.var = 'cumloss'),digits=3)

##     grcode accyear premium     1     2     3     4     5     6     7    8    9  10
##  1:     43    1988     957   133   333   431   570   615   615   615  614  614 614
##  2:     43    1989    3695   934  1746  2365  2579  2763  2966  2940 2978 2978  NA
##  3:     43    1990    6138  2030  4864  6880  8087  8595  8743  8763 8762   NA  NA
##  4:     43    1991   17533  4537 11527 15123 16656 17321 18076 18308   NA   NA  NA
##  5:     43    1992   29341  7564 16061 22465 25204 26517 27124    NA   NA   NA  NA
##  6:     43    1993   37194  8343 19900 26732 30079 31249    NA    NA   NA   NA  NA
##  7:     43    1994   46095 12565 26922 33867 38338    NA    NA    NA   NA   NA  NA
##  8:     43    1995   51512 13437 26012 31677    NA    NA    NA    NA   NA   NA  NA
##  9:     43    1996   52481 12604 23446    NA    NA    NA    NA    NA   NA   NA  NA
## 10:     43    1997   56978 12292    NA    NA    NA    NA    NA    NA   NA   NA  NA
```

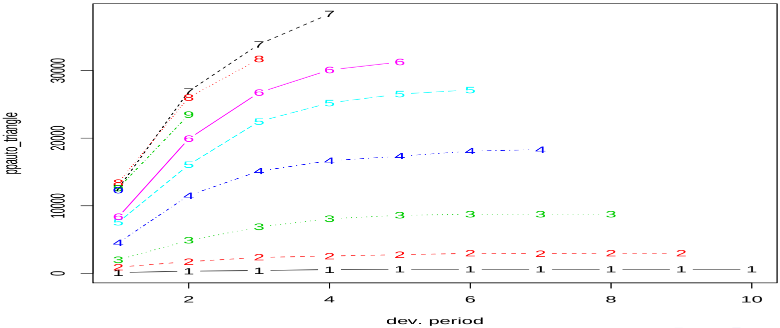

Snapshot of Loss Curves for 10 Years of
Private Passenger Auto Insurance for Single Organisation
43
353
388
620
Loss Ratio
0.0
0.5
1.0
2.5
5.0
7.5
10.0
Development Time
Cohort Year
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997


Snapshot of Loss Curves for 10 Years of
Product Liability Insurance for Single Organisation
669
683
7854
32514
Loss Ratio
0.0
0.3
0.6
0.9
2.5
5.0
7.5
10.0
Development Time
Cohort Year
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997

Chain Ladder

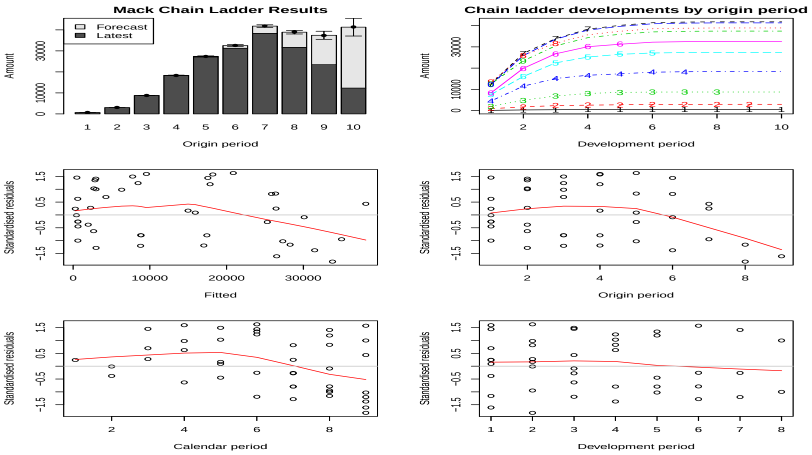
Standard R approach is ChainLadder

```
ppauto_mat <- as.matrix(dcast(ppauto_ss_dt[grcode == 43]
                             ,accyear ~ devlag
                             ,value.var = 'cumloss')[,-1,with=FALSE])

ppauto_triangle <- as.triangle(ppauto_mat)

plot(ppauto_triangle)
```

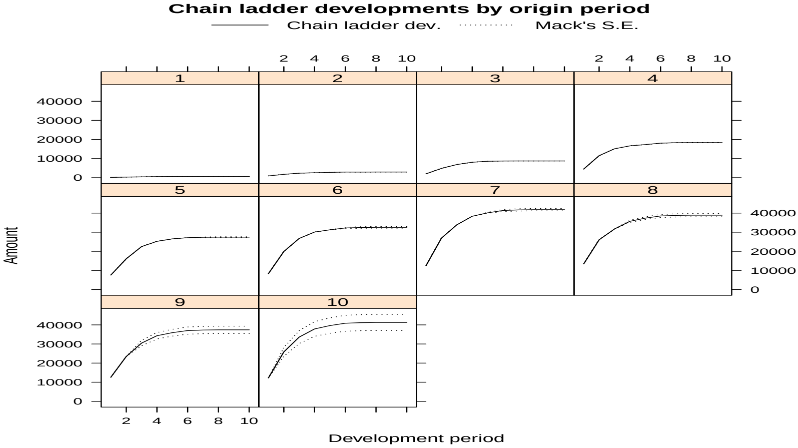


```
ppauto_mack <- MackChainLadder(ppauto_triangle, est.sigma = "Mack")

ppauto_mack$f

##  [1] 2.10486 1.29968 1.12655 1.04671 1.03069 1.00743 1.00292 1.00000 1.00000 1.00000

ppauto_mack$FullTriangle

##       dev
## origin     1     2       3       4       5       6       7       8       9      10
##      1   133   333   431.0   570.0   615.0   615.0   615.0   614.0   614.0   614.0
##      2   934  1746  2365.0  2579.0  2763.0  2966.0  2940.0  2978.0  2978.0  2978.0
##      3  2030  4864  6880.0  8087.0  8595.0  8743.0  8763.0  8762.0  8762.0  8762.0
##      4  4537 11527 15123.0 16656.0 17321.0 18076.0 18308.0 18361.5 18361.5 18361.5
##      5  7564 16061 22465.0 25204.0 26517.0 27124.0 27325.6 27405.5 27405.5 27405.5
##      6  8343 19900 26732.0 30079.0 31249.0 32208.1 32447.6 32542.4 32542.4 32542.4
##      7 12565 26922 33867.0 38338.0 40128.7 41360.4 41667.9 41789.6 41789.6 41789.6
##      8 13437 26012 31677.0 35685.7 37352.5 38499.0 38785.2 38898.6 38898.6 38898.6
##      9 12604 23446 30472.3 34328.5 35932.0 37034.8 37310.1 37419.2 37419.2 37419.2
##     10 12292 25873 33626.6 37882.0 39651.4 40868.4 41172.3 41292.6 41292.6 41292.6
```

```
plot(ppauto_mack)
```



```
plot(ppauto_mack, lattice = TRUE)
```

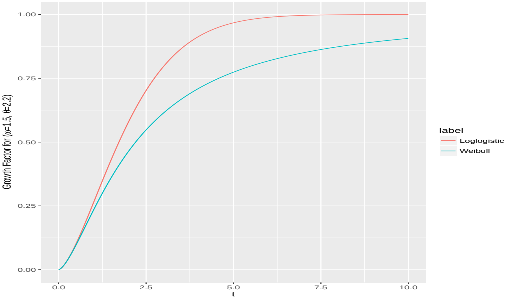


Loss Growth Modelling

Model growth cumulative losses as function
Scale losses by premium

$$g(t;\ \omega, \theta) = 1 - \exp\left(-\left(\frac{t}{\theta}\right)^{\omega}\right) \qquad \text{Loglogistic Function}$$

$$g(t;\ \omega, \theta) = \frac{t^{\omega}}{t^{\omega} + \theta^{\omega}} \qquad \text{Weibull Function}$$



Start with the Loglogistic Model

$$g(t;\ \omega, \theta) = 1 - \exp\left(-\left(\frac{t}{\theta}\right)^{\omega}\right)$$

Treat as hierarchical model - group by Accident Year

$$\text{Loss}_{\text{Y},t} \sim \text{Normal}(\mu_{\text{L},\text{Y},t}, \sigma_L)$$

where

$$\begin{aligned}
\mu_{\text{L},\text{Y},t} &= \text{LR}_\text{Y} \times \text{P}_\text{Y} \times g(t;\ \omega, \theta) \\
\sigma_\text{L} &= \text{P}_\text{Y} \times \sigma \\
\text{LR}_\text{Y} &\sim \text{Lognormal}(\mu_{\text{LR}}, \sigma_{\text{LR}})
\end{aligned}$$

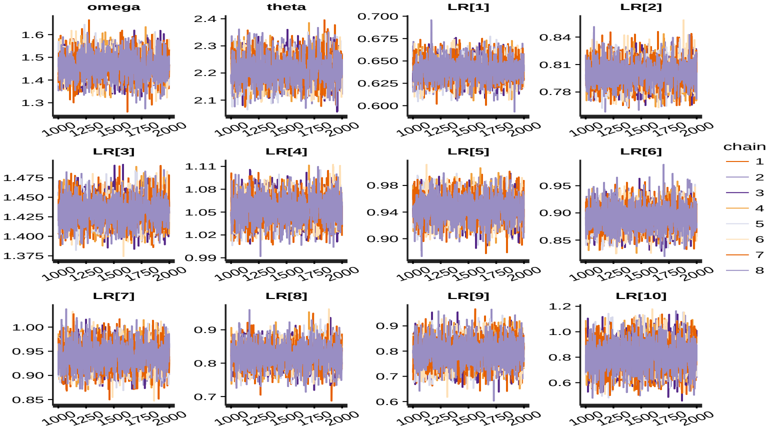
Normal prior for μ_{LR}. Lognormal prior for ω, θ, σ_{LR}, σ.

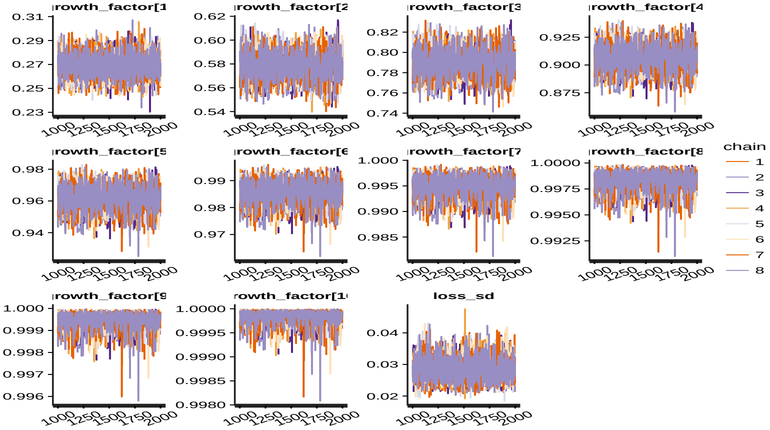
```
functions {
  real growth_factor_weibull(real t, real omega, real theta) {
    real factor;

    factor <- 1 - exp(-(t/theta)^omega);

    return(factor);
  }

  real growth_factor_loglogistic(real t, real omega, real theta) {
    real factor;

    factor <- ((t^omega) / (t^omega + theta^omega));

    return(factor);
  }
}

data {
  int<lower=0,upper=1> growthmodel_id;

  int n_data;
  int n_time;
  int n_cohort;

  int cohort_id[n_data];
  int t_idx[n_data];

  real<lower=0> t_value[n_time];

  real premium[n_cohort];
  real loss[n_data];
}
```

```
parameters {
  real<lower=0> omega;
  real<lower=0> theta;

  real<lower=0> LR[n_cohort];

  real mu_LR;
  real<lower=0> sd_LR;

  real<lower=0> loss_sd;
}

transformed parameters {
  real growth_factor[n_time];
  real loss_mean[n_cohort, n_time];

  for(i in 1:n_time) {
    if(growthmodel_id == 1) {
      growth_factor[i] <- growth_factor_weibull    (t_value[i], omega, theta);
    } else {
      growth_factor[i] <- growth_factor_loglogistic(t_value[i], omega, theta);
    }
  }

  for(i in 1:n_data) {
    loss_mean[cohort_id[i], t_idx[i]] <- LR[cohort_id[i]] * premium[cohort_id[i]] * growth_factor[t_idx[i]];
  }
}
```

```
model {
  mu_LR ~ normal(0, 0.5);
  sd_LR ~ lognormal(0, 0.5);

  LR ~ lognormal(mu_LR, sd_LR);

  loss_sd ~ lognormal(0, 0.7);

  omega ~ lognormal(0, 1);
  theta ~ lognormal(0, 1);

  for(i in 1:n_data) {
    loss[i] ~ normal(loss_mean[cohort_id[i], t_idx[i]], premium[cohort_id[i]] * loss_sd);
  }
}


generated quantities {
  real mu_LR_exp;
  real<lower=0> loss_prediction[n_cohort, n_time];

  for(i in 1:n_cohort) {
    for(j in 1:n_time) {
      loss_prediction[i, j] <- LR[i] * premium[i] * growth_factor[t_idx[j]];
    }
  }

  mu_LR_exp <- exp(mu_LR);
}
```

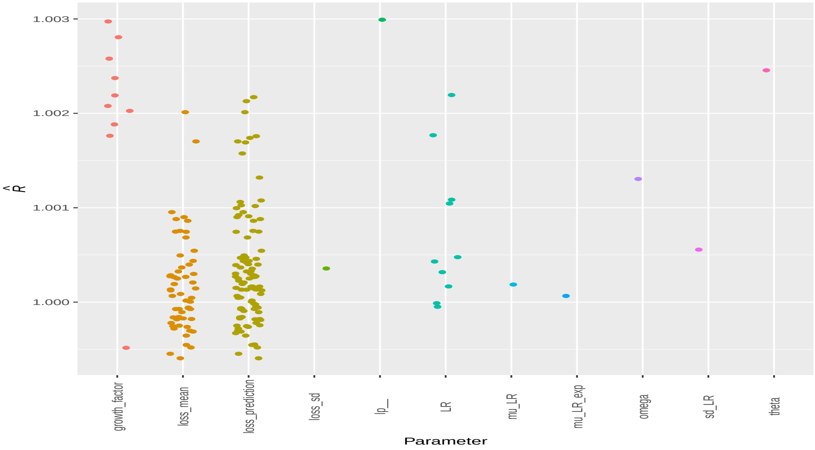
Stan Output


omega
theta
LR[1]
LR[2]
LR[3]
LR[4]
LR[5]
LR[6]
LR[7]
LR[8]
LR[9]
LR[10]
1000 1250 1500 1750 2000
chain
1
2
3
4
5
6
7
8


rowth_factor[1
rowth_factor[2
rowth_factor[3
rowth_factor[4
rowth_factor[5
rowth_factor[6
rowth_factor[7
rowth_factor[8
rowth_factor[9
rowth_factor[10
loss_sd
chain
1
2
3
4
5
6
7
8
1000 1250 1500 1750 2000

Check simple diagnostics:

```
## Warning:  Removed 45 rows containing missing values (geom_point).
```



Model Creation

Model Iteration

Posterior Predictive Checks

Conclusions

Get In Touch

Mick Cooney
michael.cooney@applied.ai

Slides and code available on BitBucket:
https://www.bitbucket.org/kaybenleroll/dublin_r_workshops