Loss Simulation Model: Testing and fitting

Joseph O. Marker
Marker Actuarial Services, LLC
and University of Michigan
CLRS 2010 Meeting

Expected vs Actual Distribution

- Test distributions of:
  - Number of claims (frequency)
  - Size of ultimate loss (severity)

- Sources of significant difference between actual and expected amounts:
  - Programming or communication errors
  - Not understanding how statistical language (e.g. "R") works.
  - Errors or misleading results in "R".

Display Raw Simulator Output

- Claims file

Simulation No	Occurrence No	Claim No	Accident Date	Report Date	Line	Type
1	1	1	20000104	20000227	1	1
1	2	1	20000105	20000818	1	1
....						

- Transactions file

Simulation No	Occurrence No	Claim No	Date	Trans-action	Case Reserve	Payment
1	1	1	20000227	REP	2000	0
1	1	1	20000413	RES	89412	0
1	1	1	20000417	CLS	-91412	141531
……..	……….	……..	………			

Another use for Testing information

- Create Ultimate Loss File for Analysis – Layout

Simula -tion. No	Occur-rence No	Claim No	Accident. Date	Report. Date	Line	Type	Case. Reserve	Pay-ment

- Idea: Another use for this section of paper
  - If an insurer can summarize its own claim data to this format, then it can use the tests we will discuss to parameterize the Simulator using its data.
  - We have included in this paper all the "R" code used in testing.

Emphasis in the Paper

- Document the "R" code used in performing various tests.
- Provide references for those who want to explore the modeling more deeply.
- Provide visual as well as formal tests
  - QQPlots, histograms, densities, etc.

Test 1 – Frequency, Zero-Modification, Trend

- Model parameters:
  - # Occurrences ~ Poisson (mean = 120 per year)
  - 1,000 simulations
  - One claim per occurrence
  - Frequency Trend 2% per year, three accident years
  - Pr[Claim is Type 1] = 75%;   Pr[Type 2] = 25%
  - Pr[CNP(“Closed No payment”)] = 40%
  - “Type” and “Status” independent.
  - Status is a category variable for whether a claim is closed with payment.

- Test output to see if its distribution is consistent with assumptions.

Test 1 – Classical Chi-square

Contingency Table

	Actual Counts					Expected Counts		
	Type 1	Type 2	Margin			Type 1	Type 2	Margin
CNP	111,066	37,007	0.398906		CNP	111,029.0	37,044.0	0.398906
CWP	167,268	55,857	0.601094		CWP	167,305.0	55,820.0	0.601094
Margin	0.749826	0.250174	371,198			0.749826	0.250174	371,198

$$X^2 = \sum_i \sum_j \frac{(Actual_{ij} - Expected_{ij})^2}{Expected_{ij}} = 0.0819$$

Pr [$X^2 > 0.0819$] = 0.775. The independence of Type and Status is supported.

Test 1 – Regression approach

- Previous result can be obtained using **`xtabs`** command in "R"
- Result can also be obtained using Poisson GLM
  - Full model:

```
model6x<- glm(count ~  Type + Status + Type*Status,
   data = temp.datacc.stack, family = poisson, x=T)
```

  - Reduced model:

```
model5x<- glm(count ~  Type + Status ,
   data = temp.datacc.stack, family = poisson, x=T)
```

- Independence obtains if the interactive variable Type*Status is not significant.

Test 1 – Analysis of variance

- **`anova( model5x, model6x, test="Chi")`**

```
Analysis of Deviance Table
Response: count
           Terms Resid. Df   Resid. Dev         Test Df
1                + Type + Status    143997   160969.366
2 Type + Status + Type * Status    143996   160969.284 +Type:Status  1

      Deviance     Pr(Chi)
1
2 0.0819088429 0.774727081
```

- Result matches the previous X^2 Test.

- We did not show here the model coefficients, which will produce the expected frequency for each combination of Type and Status.

Test 2 – Univariate size of loss

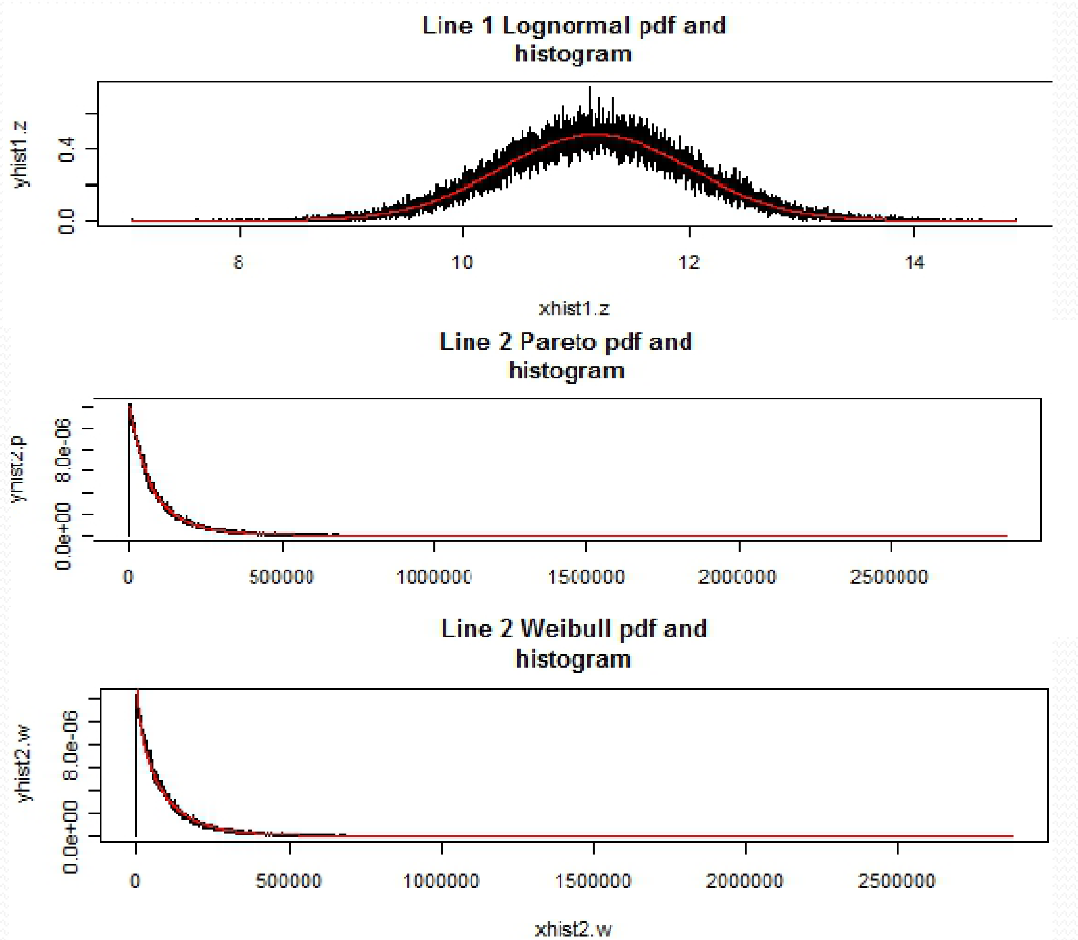
- Model parameters:
  - Three lines – no correlation in frequency by line
  - # Claims for each line ~ Poisson (mean = 600 per year)
  - Two accident years, 100 simulations
  - Size of loss distributions
    - Line 1 – lognormal
    - Line 2 – Pareto
    - Line 3 -- Weibull
  - Zero trend in frequency and size of loss.

- Expected count = 600 (freq) x 100 (# sims) x 3 (lines) x 2 (years) = 360,000.
- Actual # claims: 359,819.

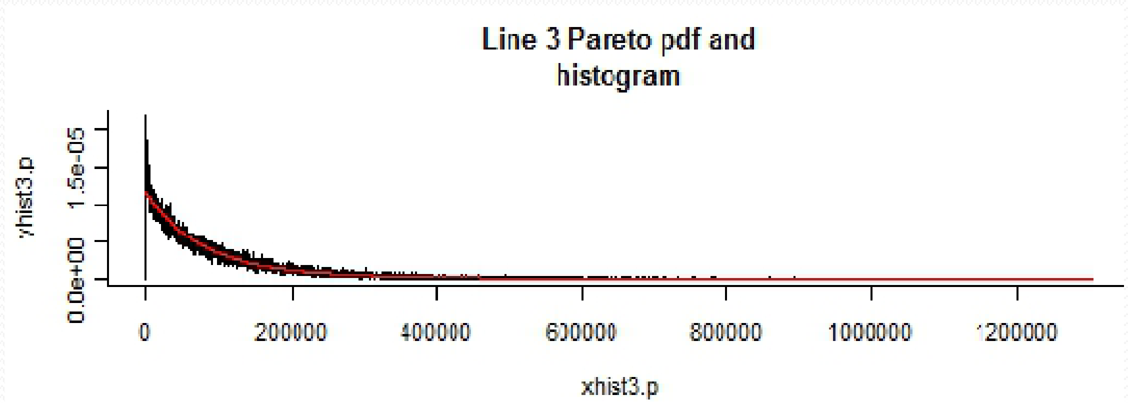
Size of loss – testing strategy

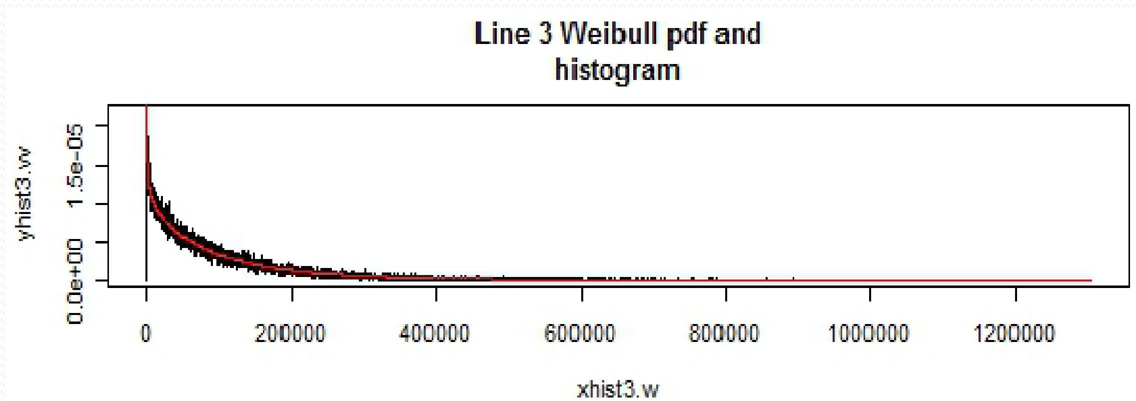
- Person doing testing $\neq$ Person running simulation.
- Test all three distributions on each line’s output.
- Produce plots to “get a feel” for distributions.
- Fit using maximum likelihood estimation.
- Produce QQ (quantile-quantile) plots
- Run formal goodness-of-fit tests.

Size of loss – Histograms and p.d.f.



Size of loss – Histograms and p.d.f.





Size of loss

- The plots above compare:
  - Histogram of empirical distribution
  - Density of the theoretical distribution with m.l.e. parameters
- The plots show that both Weibull and Pareto fit Lines 2 and 3 well.
- QQ plots offer another perspective.

Size of loss – QQ Plots

- Example of "R" code to produce a QQ Plot

```
thqua.w2 <-
  rweibull(n2,shape=fit.w2$estimate[1],scale=fit.w2$estimate[2])
```

generate a random sample same size n2 as empirical data

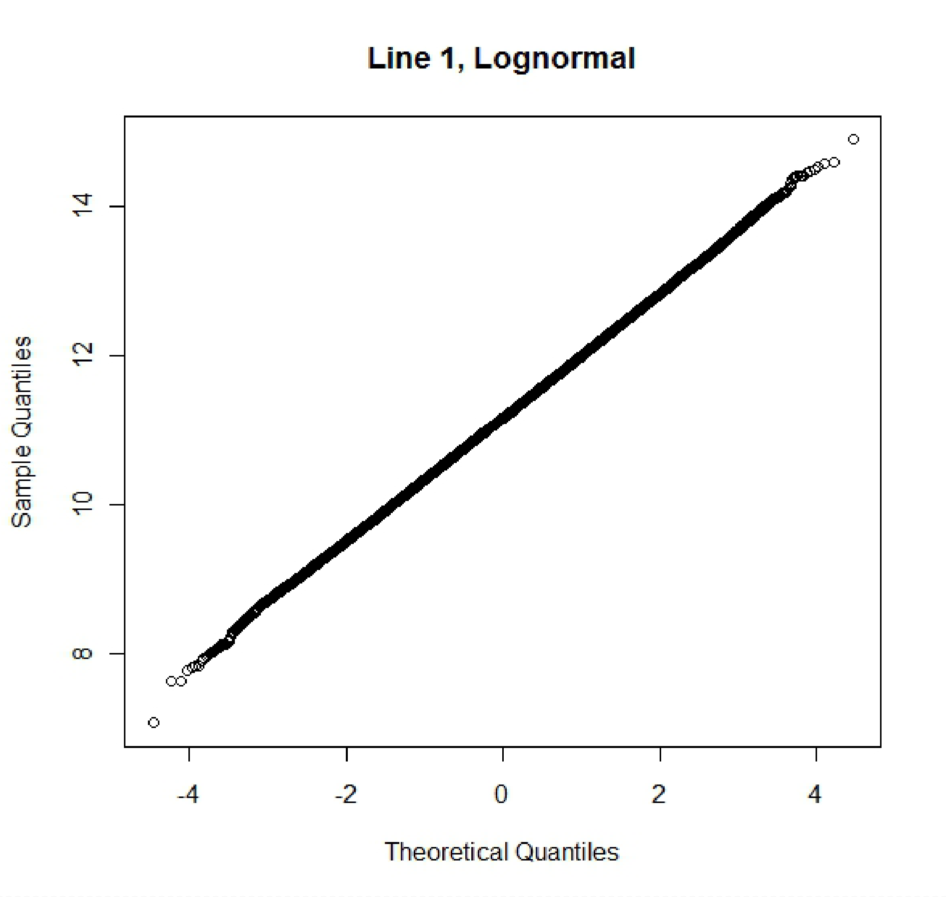
```
qqplot(ultloss2,thqua.w2,xlab="Sample Quantiles",
           ylab="Theoretical Quantiles", main="Line 2, Weibull")
```

ultloss2 is empirical data, thqua.w2 is the generated sample

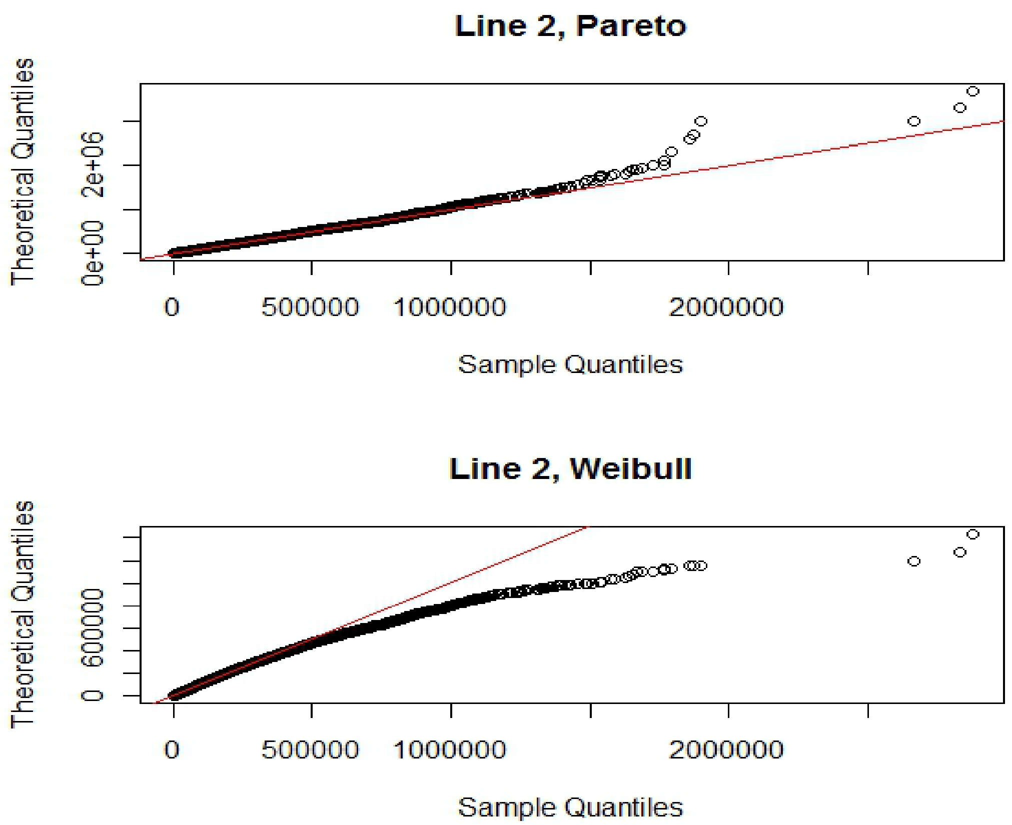
```
abline(0,1,col="red")
```

- One can also replace the sample with the quantiles of the theoretical Weibull c.d.f.

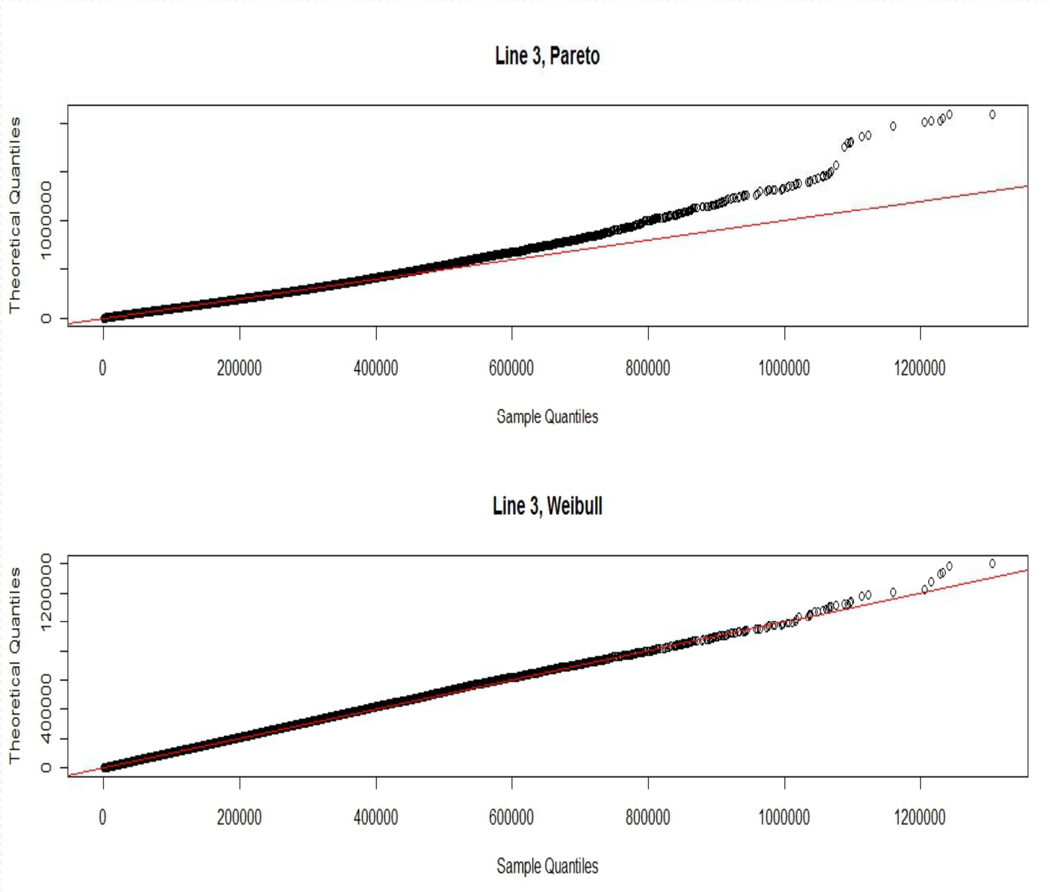
Size of Loss – QQ Plot, Line 1



Size of Loss – QQ Plot, Line 2



Size of Loss – QQ Plot, Line 3.



Size of Loss – Fitted distributions

- From QQ Plots, it appears that lognormal fits Line 1, Pareto fits Line 2, and Weibull fits Line 3.

- Chi-square is a formal goodness-of-fit test. Section 6 discusses setting up the test for Pareto on Line 2. Appendix B contains "R" code for all the chi-square tests.

- Komogorov-Smirnov test was applied also, but too late to include results in this presentation.

Size of Loss – Chi-square g.o.f. test

Setting up bins and the expected and actual # claims by bin is not easy in R.

Define break points and bins:

```
s = sqrt(var(ultloss2))
ult2.cut <- cut(ultloss2.0,                                    ##binning data
  breaks = c(0,m-s/2,m,m+s/4,m+s/2,m+s,m+2*s,2*max(ultloss2)))
     Note:  ultloss2.0 is vector of loss sizes, m = mean
```

```
The table of expected and observed values by bin:
#          E.2   O.2     x.sq.2
#[1,] 43993.890 44087 0.19705959     Notes:
#[2,] 35651.989 35680 0.02200752       E.2 expected number
#[3,] 10493.758 10323 2.77864169       O.2 actual number
#[4,]  7240.583  7269 0.11152721       x.sq.2 Chi-sq statistic
#[5,]  9277.383  9164 1.38570182
#[6,]  8063.576  8176 1.56743997
#[7,]  5289.820  5312 0.09299630
```

Size of Loss – Chi-square g.o.f. test

- Execute the Chi-Square test

```
df=length(E.2)-1-2            ## degrees of freedom Result= 4
chi.sq.2 <- sum(x.sq.2)       ## test statistic  Result = 6.155374
qchisq(.95,df)                ## critical value Result = 9.487729
1-pchisq(chi.sq.2,df)         ## p-value  Result = 0.1878414
```

- Important – degrees of freedom = 4, not 6, because the two parameters for expected distribution were determined from m.l.e. on the data rather than from a predetermined distribution.
- Using the chi-squared test in R directly would produce a wrong *p*-value:

```
chisq.test(O.2,p=E.2/n2.0)
```

This test uses degrees of freedom = 6

Correlation

- Model allows correlated variables in two ways:
  - Frequencies among lines.
  - Report lag and size of loss.

- We tested the correlation feature for frequency by line.
  - To do this, first specify the parameters for Poisson or negative binomial frequency by line.

  - Then specify correlation matrix and the copula that links the univariate frequency distributions to the multivariate distribution.

- The correlation testing helped the programmer determine how the copula statements from “R” actually work in the model.

Correlation – simulation parameters

- Simulator was run 7/20/2010 with parameters:
  - Three lines
  - Annual frequency by line is Poisson with mean 96.
  - One accident year.
  - 1,000 simulations
  - Gaussian (normal) copula
  - Frequency correlation matrix:

Correlation	Line 1	Line 2	Line 3
Line 1	1	0	0.99
Line 2	0	1	-0.01
Line 3	0.99	-0.01	1

Correlation – data used

- The annual number of claims were summarized by simulation and line to a file "D:/LSMWP/byyear.csv".

- Visualize this data:

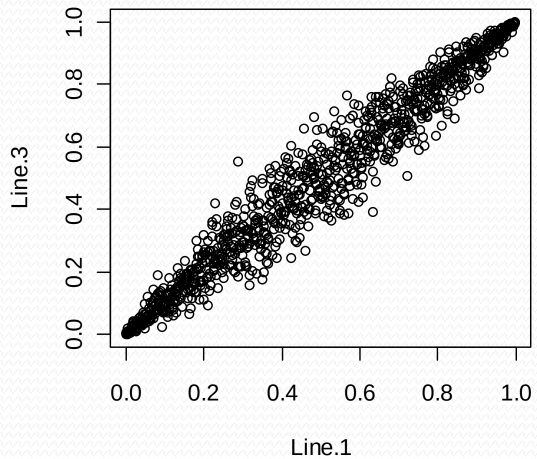
Row (simulation)	Line 1	Line 2	Line 3
1	114	95	117
2	89	85	90
….	….	….	….
99	103	78	101
100	96	106	99

Correlation – Fitting data

- Detail of statistical testing for correlation is in section 6.2.3 and Appendix B of the paper.

- Data was fit to normal copula using both m.l.e. and inversion of Kendall's tau, using all 1,000 observations, and then goodness of fit tests were applied to each pair of lines.

- Scatter-plot of Line 1 and Line 3 data



Correlation – estimated correlation from data

- Details of maximum likelihood estimate of correlations

	Estimate	Std. Error	z value	Pr(>|z|)
Rho(line 1 & 2)	-0.002112605	0.031977597	-0.06606516	0.9473259
Rho(line 1 & 3)	0.979258746	0.000921392	1062.80366235	0.0000000
Rho(line 2 & 3)	-0.010486832	0.031974114	-0.32797880	0.7429277

- Example of statements used for first "rho" above:

```
normal2.cop <- normalCopula(c(0),dim=2,dispstr="un")
gofCopula(normal2.cop, x12, N=100, method = "mpl")
```

Note: x12 is a dataset without line 3 observations.

Correlation – goodness of fit

- The empirical copula and hypothesized copula are compared under the null hypothesis that they are from the same copula. Cramér-von-Mises ("CvM") statistic S_n is used.

- Goodness of fit test runs very slowly, so each pair of lines were compared using only the first 100 simulations.

- The two-sample Kolmogorov-Smirnov test was performed. This compared the empirical distribution with a random sample from the hypothesized distribution.

Correlation – g.o.f. results

- Line 1&2
  - Parameter estimate(s): -0.002100962
  - Cramer-von Mises statistic: 0.0203318 with *p*-value 0.4009901
-
- Line 1&3
  - Parameter estimate(s): 0.97926
  - Cramer-von Mises statistic: 0.007494245 with *p*-value 0.3811881
-
- Line 2&3
  - Parameter estimate(s): -0.01049841
  - Cramer-von Mises statistic: 0.01614539 with *p*-value 0.5891089

Final Thoughts on Testing

- Initial tests were simple because we were also checking the mechanics of the model.
- There are many more features of the model to explore and to test.
- The testing statements can also be applied to parameterize the model using an insurer's data.
- The tests described only test ultimate distributions, not the loss development patterns.