Pooling and Insurance Actuarial Techniques - Pooling and Insurance¹

Professor Benjamin Avanzi



04 January 2022



- Pooling and Insurance
- 2 Risk pooling in case of independent risks
- 3 Risk pooling in case of correlated risks
- Basic risk transfer arrangements
- Seferences



- Pooling and Insurance
- 2 Risk pooling in case of independent risks
- 3 Risk pooling in case of correlated risks
- 4 Basic risk transfer arrangements
- 6 References



Pooling and Insurance

- In the following sections we illustrate the effect of pooling, and then insurance, on the level of risk for an individual.
- We will follow the following steps:
 - Define a situation, where risks are independent
 - Explore possible outcomes from simulation, comparing the cases without and with pooling
 - Analyse that situation mathematically
 - Extend this to a situation where risks are dependent
 - Contrast a pure pool with an insurance company, and explain how insurance adds to pooling
- The central question is: What is the basic idea behind insurance?



- Pooling and Insurance
- 2 Risk pooling in case of independent risks
- 3 Risk pooling in case of correlated risks
- 4 Basic risk transfer arrangements
- 5 References





Risk pooling in case of independent risks

- Assumptions
- Example with simulations
- Example with formulas
- With pooling



Independence

Independence:

- Means information about one variable carries no information about the other (F(x,y)=F(x)F(y))
- Is sometimes a reasonable assumption.
- For instance: the risk of dying for most individuals is an independent risk since the chance that a person dies does not usually depend on whether or not someone else has died
- There are exceptions, usually related to extreme events / disasters (such as a pandemic)



Assumptions for the illustration

Risks:

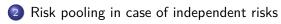
- n independent risks X_i for i = 1 to n (e.g., n individuals)
- Each risk a loss of L with probability q or no loss with probability (1-q) (e.g. life insurance)
- Each risk can incur only one loss

$$Pr[X_i = L] = q$$

$$Pr[X_i = 0] = (1 - q)$$

(e.g. one can die only once)





- Assumptions
- Example with simulations
- Example with formulas
- With pooling



One possible year with n = 100, q = 0.01, L = 10000

```
set.seed(1)
claims1 <- 10000 * rbinom(100, 1, 0.01)
print(claims1, width = 55)
##
      [1]
                                                     0
                                                            0
##
      [9]
                                                     0
                                                            0
##
     [17]
                  10000
                                                     0
                                                            0
     Γ251
##
                       0
                              0
                                                     0
                                                            0
     [33]
##
                              0
                                                     0
                                                            0
     [41]
##
                              0
                                      0
                                                     0
                                                            0
     Γ491
##
                              0
                                                     0
                                                            0
     [57]
                       0
                              0
##
                                      0
                                                     0
                                                            0
     [65]
##
                              0
                                                     0
                                                            0
     [73]
                       0
                              0
                                                            0
##
                                      0
                                             0
                                                     0
     [81]
##
                              0
                                             0
                                                     0
                                                            0
     [89]
                       0
                                                            0
##
               0
                              0
                                      0
                                             0
                                                     0
                                                                   0
     [97]
                0
                       0
                              0
                                      0
##
```



```
claims2 \leftarrow 10000 * rbinom(100, 1, 0.01)
print(claims2, width = 55)
##
      [1]
                                   10000
                                                       0
                                                               0
##
      [9]
                                0
                                        0
                                                0
                                                       0
                                                               0
##
     [17]
                                0
                                          10000
                                                       0
                                                               0
##
     [25]
                                0
                                                0
                                                       0
                                                               0
     [33]
##
                                0
                                                0
                                                       0
                                                               0
     [41]
                                0
##
                                                       0
                                                               0
     [49]
##
                                0
                                        0
                                                0
                                                       0
                                                               0
     Γ571
##
                        0
                                0
                                                0
                                                       0
                                                               0
                                                                       0
     [65]
                0
                        0
                                0
                                        0
                                                0
                                                               0
                                                                       0
##
                                                       0
     [73]
                        0
##
                                0
                                        0
                                                0
                                                       0
                                                               0
                                                                       0
     [81]
                0
                        0
                                0
                                        0
                                                0
                                                       0
                                                               0
                                                                       0
##
     [89]
                        0
                                0
                                        0
                                                       0
                                                               0
##
                                                0
                                                                       0
     [97]
                0
                        0
                                0
                                        0
##
```





##

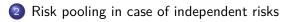
##

```
claims4 <- 10000 * rbinom(100, 1, 0.01)
print(claims4, width = 55)
##
      [1]
                                                 10000
                                                              0
##
      [9]
                               0
                                                      0
                                                             0
##
     [17]
                               0
                                                      0
                                                                10000
##
     [25]
                               0
                                                      0
                                                              0
                                                                     0
     [33]
##
                               0
                                       0
                                              0
                                                      0
                                                              0
     [41]
                               0
##
                                                      0
                                                              0
     [49]
##
                               0
                                       0
                                              0
                                                      0
                                                              0
     Γ571
##
                               0
                                       0
                                              0
                                                      0
                                                              0
     [65]
                0
                        0
                               0
                                       0
                                              0
                                                              0
                                                                     0
##
                                                      0
     [73]
                        0
##
                               0
                                       0
                                              0
                                                      0
                                                              0
                                                                     0
     [81]
                0
                        0
                               0
                                       0
                                              0
                                                      0
                                                              0
                                                                     0
##
     [89]
                        0
                               0
                                       0
                                              0
                                                      0
                                                              0
##
                                                                     0
     [97]
                0
                        0
                               0
                                       0
##
```



```
claims5 <- 10000 * rbinom(100, 1, 0.01)
print(claims5, width = 55)
##
      [1]
                                                              0
                                                      0
##
      [9]
                          10000
                                                      0
                                                              0
##
     [17]
                               0
                                                      0
                                                              0
##
     [25]
                               0
                                                      0
                                                              0
     [33]
                               0
##
                                       0
                                              0
                                                      0
                                                              0
     [41]
                               0
##
                                                      0
                                                              0
     [49]
##
                               0
                                       0
                                              0
                                                      0
                                                              0
     Γ571
##
                               0
                                              0
                                                      0
                                                              0
                                                                     0
##
     [65]
                0
                        0
                               0
                                       0
                                              0
                                                      0
                                                              0
                                                                     0
     [73]
                        0
##
                               0
                                       0
                                              0
                                                      0
                                                              0
                                                                     0
     [81]
                0
                        0
                               0
                                       0
                                              0
                                                      0
                                                              0
                                                                     0
##
     [89]
                        0
                               0
                                       0
                                              0
                                                      0
                                                              0
                                                                     0
##
     [97]
                0
                        0
                               0
                                       0
##
```





- Assumptions
- Example with simulations
- Example with formulas
- With pooling



In the example above

An individual has a 0.01 chance of dying and the insurer will pay \$10,000 upon their death. Calculate the expected death benefit and variance of the death benefit for this individual.

Expected benefit

$$0.01 \times 10,000 = 100$$

Variance

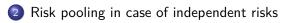
$$0.01 \times 0.99 \times 10,000^2 = 990,000$$

• Standard deviation is $\sqrt{990,000} = 995$.

This corresponds to the moments of this unique risk. Of course, insurers have more than one risk in their portfolio.

This leads to pooling.





- Assumptions
- Example with simulations
- Example with formulas
- With pooling



Risk Pooling - How does it work?

- Pool the n risks
- Each individual agrees to pay the actual average of the total losses into the pool
- The pool pays the losses of the members of the pool who incur a loss
- Average of the losses

$$A = \frac{\sum_{i=1}^{n} X_i}{n}$$

with $X_i = L$ if a loss occurs or $X_i = 0$ if no loss occurs.

This is a random variable - we do not know what it will be until the losses occur - The good luck of some compensates for the bad luck of others.



Simulated example

```
In this case, the hundred individuals all pay the same amount,
corresponding to the average. For the 5 cases, this is
print(c(mean(claims1), mean(claims2), mean(claims3), mean(claims4),
    mean(claims5)))
```

```
## [1] 100 200   0 200 100
```



Expectation with pooling

Expected value of the average loss for the pool

$$E[A] = \mu_A = E\left[\frac{\sum_{i=1}^n X_i}{n}\right]$$

Number of losses (in this case) has a Binomial (n, q) distribution, so that probability that j individuals incur a loss is

$$\Pr\left(\sum_{i=1}^n X_i = jL\right) = \binom{n}{j} q^j \left(1-q\right)^{n-j} \quad j = 0, 1, 2 \dots, n.$$



Expected value of the average loss

$$\mu_{A} = E\left[\frac{\sum_{i=1}^{n} X_{i}}{n}\right]$$

$$= \sum_{j=0}^{n} \frac{jL}{n} \binom{n}{j} q^{j} (1-q)^{n-j}$$

$$= \frac{L}{n} \sum_{j=0}^{n} j \binom{n}{j} q^{j} (1-q)^{n-j}$$

$$= \frac{L}{n} nq = qL$$

$$= \mu_{i}$$

Expected value of the average loss in the pool is the same as the expected value of the individual loss!



Note - Expectation is a linear operator

We have

$$E[k] = k$$

where k is a constant, and

$$E[aX + bY + c] = aE[X] + bE[Y] + c$$

where X and Y are random variables that need not be independent, and a, b and c are constants



Revision - Properties of Variance

For any random variable X

$$Var\left[X\right] = E\left[X^2\right] - E\left[X\right]^2$$

so that

$$E\left[X^2\right] = Var\left[X\right] + E\left[X\right]^2$$

and also

$$Var[kX] = k^2 Var[X]$$

Also,

$$Var(X + Y) = Var(X) + Var(Y)$$

if X and Y are independent.



Variance with pooling

$$\sigma_A^2 = Var \left[\frac{\sum_{i=1}^n X_i}{n} \right]$$

$$= \frac{1}{n^2} Var \left[\sum_{i=1}^n X_i \right]$$

$$= \frac{1}{n^2} \sum_{i=1}^n Var [X_i] \quad \text{because the } X_i \text{'s are independent}$$

$$= \frac{n}{n^2} Var [X_i] \quad \text{because the } X_i \text{'s have same distribution}$$

$$= \frac{n}{n^2} L^2 Var [Y_i] \quad \text{where } Y_i \text{ is Bernoulli}(q)$$

$$= \frac{L^2}{n} q(1-q)$$



Risk Pooling - Pool Variance in Words

The variance of the average of the pool losses is the variance of each individual loss $q(1-q)L^2$ divided by the number of individuals in the pool

$$\sigma_A^2 = \frac{q(1-q)L^2}{n} = \frac{Var(X)}{n}$$



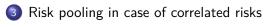
Risk Pooling - So Why Pool?

- For risk averse individuals, the "happiness" of each individual who participates in the risk pooling is higher (variability is lower)
- A mathematical model for explaining this, in microeconomics, would use so-called "utility functions"
- If they fully insure, they exchange all their risk, otherwise they exchange only a proportion of it
- If the risk bearer had a choice of how much risk to transfer (e.g. an insurer vs an reinsurer, choice of excess on an insurance contract), this optimal proportion would depend on the cost of transfer and their risk aversion.



- Pooling and Insurance
- Risk pooling in case of independent risks
- 3 Risk pooling in case of correlated risks
- 4 Basic risk transfer arrangements
- 6 References





- Assumptions
- Pooling of Correlated Risks
- Effect of pooling
- Diversification benefits



Assumptions

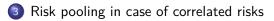
- Here we relax the assumption of independence, as this is a major assumption
- We consider the simplest dependence model possible: correlation
- The basic formulas about correlation are

$$Var(X + Y) = Var(X) + Var(Y) + 2Cov(X, Y),$$

and then the coefficient of correlation is defined as

$$-1 \le \rho(X, Y) = \frac{Cov(X, Y)}{\sqrt{Var(X)}\sqrt{Var(Y)}} \le 1.$$





- Assumptions
- Pooling of Correlated Risks
- Effect of pooling
- Diversification benefits



Pooling of Correlated Risks

Simplest case

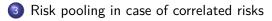
- Two losses X and Y each with the same expected value μ and the same variance σ^2
- Correlation $\rho(X, Y) = \rho$, equal variances with $\sigma_X = \sigma_Y = \sigma$. Hence $Cov(X, Y) = \rho \sigma^2$.

Average expected loss is

$$E\left[\frac{X+Y}{2}\right] = \frac{E[X+Y]}{2} = \frac{E[X]+E[Y]}{2} = \frac{2\mu}{2} = \mu$$

which is what we expected.





- Assumptions
- Pooling of Correlated Risks
- Effect of pooling
- Diversification benefits



Effect of pooling

Variance of the average loss is

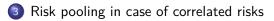
$$Var\left[\frac{X+Y}{2}\right] = \frac{1}{4}\left[\sigma^2 + \sigma^2 + 2\rho\sigma^2\right]$$
$$= \frac{\sigma^2}{2}(1+\rho)$$

- Pooling of correlated risks multiplies the variance, relative to the variance if the risks were independent, by the factor $(1 + \rho)$ where $-1 \le \rho \le 1$.
- ullet If risks are perfectly correlated (ho=1), the variance is

$$Var\left[\frac{X+Y}{2}\right] = \sigma^2$$

Then there is no apparent benefit in pooling.





- Assumptions
- Pooling of Correlated Risks
- Effect of pooling
- Diversification benefits



Diversification benefits

As soon as dependence is not perfect, there are diversification benefits:

diversification benefit
$$= \sigma^2 - \frac{\sigma^2}{2} (1 + \rho)$$

 $= \sigma^2 \frac{1 - \rho}{2}$

The more the diversification benefits, the lower the risk, and the lower the capital requirements. Accurate calculation of diversification benefits is difficult but crucial.



- Pooling and Insurance
- Risk pooling in case of independent risks
- 3 Risk pooling in case of correlated risks
- Basic risk transfer arrangements
- 5 Reference



Basic risk transfer arrangements

Essentially, there are two possible risk transfer arrangements:

- Mutual company / Friendly society / simple pool: Individuals exchange (some of) the individual risk X_i for (some of) the less variable average risk of the total pool $\frac{\sum_{i=1}^{n} X_i}{n}$
 - Calculations in this section assume we are in this situation.
- Insurance: Individuals exchange (some of) the individual risk X_i for a known, deterministic premium. The shareholders cover the downside risk by making capital available, and expect remuneration for it. Hence, this is generally more expensive (but less risky). In this case, the premium would be exactly qL=100, plus loadings (e.g. cost of capital). In any case, it is **fully deterministic**, in contrast to the former arrangement.

Historically, the former arrangement above appeared first, and "modern" insurance as we know it developed only later.

- Pooling and Insurance
- Risk pooling in case of independent risks
- Risk pooling in case of correlated risks
- 4 Basic risk transfer arrangements
- 6 References



References I

Atkinson, M. E., and David C. M. Dickson. 2011. *An Introduction to Actuarial Studies*. 2nd ed. Edward Elgar.

