H. Petersson and C. Wohlin, "Evaluating Defect Content Estimation Rules in Software Inspections", Proceedings EASE: Empirical Assessment and Evaluation in Software Engineering, Keele, UK, 2000.

Evaluating Defect Content Estimation Rules in Software Inspections

Håkan Petersson and Claes Wohlin
Dept. of Communication Systems, Lund University
Box 118, SE - 221 00 Lund, Sweden
Fax: +46-46-145823, E-mail: (hakan.petersson, claes.wohlin)@telecom.lth.se

Abstract

This paper is concerned with evaluating two different improvements of an existing defect content estimation model. The model improved is a curve-fitting model. Two new estimation rules are evaluated and compared with the original model. Further, the new estimation rules and the original model are evaluated against one of the most successful defect content estimation models, which is a capture-recapture model. It is concluded that one of the new estimation rules for the curve-fitting model could be a good complement to the capture-recapture model. Moreover, it is concluded that the results support previously published results and hence show strong evidence for that the studied model are mature enough to be transferred to industrial use to support continuous quality assessment and control.

1. Introduction

Continuous quality assessment and control is important throughout the software life cycle. Methods to estimate the remaining number of faults after an inspection provide interesting opportunities of gaining control of the faults in the software up-front, instead of having unpleasant surprises in testing. The opportunities include providing an input for decision making so that managers may decide whether to continue, re-inspect or re-work the inspected artifact. This would, hopefully, lead to fault control from the requirements specification to release.

The basic idea behind defect content estimation from software inspections is to use the overlap and non-overlap between individual inspectors to make an estimate. The methods come historically from estimation of animal population [Otis78]. Most methods used fall into the category of capture-recapture, which basically means, in software engineering terms, that a fault is caught by one inspector and if another inspector has found the same fault, it is said to be recaptured. Several different types of estimators can be used to actually perform the estimation. The estimators are based on different assumptions. Another type of models is curve-fitting models, which based on plotting the data in a certain way allow for estimation of the defect content. Curve-fitting models are further discussed in Section 2.1.

The objective in this paper is to evaluate two modifications of the estimation rule for a curve-fitting model denoted DPM (Defect Profile Method), which was first introduced in [Wohlin98]. The estimation from the model is based on a rule. Two different ways of modifying this rule are evaluated through an empirical study using 30 data sets from inspections. The new estimation rules together with the original DPM are compared with one of the most successful capture-recapture models in software engineering [Briand98, Miller99, Petersson99b], namely the Jackknife estimator.

It is shown that improvements are possible although it is difficult to generalise the results. It is concluded that the Jackknife estimator can be recommended for industrial use, although using an experience-based approach as a complement could be wise. Finally, it is concluded that the study to a large extent supports previous results in the field and that this is important because it forms a good basis for transferring the models to industrial use as a means for quality assessment and control throughout the software life cycle.

2. Improving the Detection Profile Method

2.1. The Detection Profile Method

The Detection Profile Method, DPM, is introduced in [Wohlin98]. Compared to the traditional capture-recapture methods DPM takes a different approach of estimating the number of remaining faults after inspection. DPM utilises curve fitting to produce its estimate. The estimation procedure is explained in detail in [Wohlin98], though in short it can be summarised as the following; the values of how many reviewers found each fault are sorted decreasingly and an exponential function, $y = a \cdot \exp(b \cdot x)$, is fitted to them. The estimation is taken as the value where the curve intercept y = 0.5.[1] This is referred to as an estimation rule. An example of the use of DPM is shown in Figure 1.

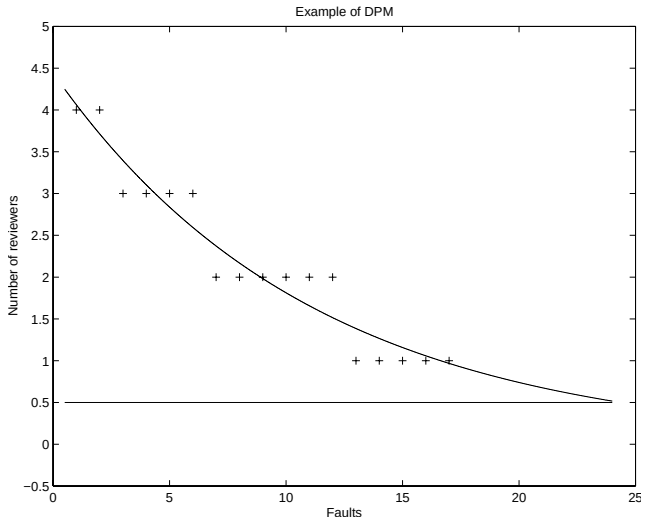

Figure 1. Example of DPM estimation.

The DPM has been examined in a couple of different studies. In [Wohlin98], where DPM was introduced, it was found to perform best of the three estimators studied. Other studies [Briand98, Petersson99b] have also applied the DPM successfully although it has been noticed that it has a tendency to underestimate.

2.2. Improvements of DPM

Since some studies show that DPM has a tendency to underestimate, one step towards improvement is to decrease the number of underestimations. One way of improving this tendency is to make some sort of bias correction. That is, to try to increase all esti-

1. If this value is lower than the number of data points, the number of data points is used instead.

mates given by the DPM in order to increase the balance between underestimations and overestimations. A study of bias corrections and their application for capture-recapture methods can be found in [Petersson99b]. When considering DPM, there is one parameter that may be modified in order to increase the estimates. As described in Section 2.1 the estimate given by DPM is taken when the fitted curve intercepts the value of y = 0.5. If this estimation parameter 0.5 is lowered, it results in forcing the interception to the right, i.e. giving higher estimates.

As can be seen in [Petersson99a], adding more inspectors leads to less bias. This leads to the conclusion that the estimation parameter should have different values depending on the number of participating inspectors.

Even if lowering the interception level will raise the bias and manage to produce a more balanced mix between underestimations and overestimations there are still problems. Since the change of the estimation parameter affects all the estimations the already high outlier estimations would become even higher. Moreover, the curve being a decreasing exponential curve, an equally sized step on the y-axis gives a larger impact on the x-axis the lower we get on the y-axis. This means we should expect larger deviations for the estimates if lowering the estimation parameter of 0.5. The question is whether the gain in bias is worth the increase in deviation. This creates a new estimation rule, where the parameter is dependent on experience from a fit data set and the number of inspectors. This modified version of DPM is denoted DPM_{mod}.

Another situation occurs when most faults are found by most of the inspectors and few faults and found by single inspectors. This occurs when most faults are fairly easy to find. For example, this will lead to, with four inspectors, that many of the faults will be found by either three or four inspectors and few by only one or two. However, this situation leads to a fitted curve that does not cross the 0.5 level until very late. This contradicts the basic assumption behind the capture-recapture model, i.e. that a large overlap among the inspectors leads to few faults left to be found and vice versa.

One approach in order to increase the bias but that also will avoid the above described situation, is to use the derivative of the fitted curve as an estimation rule. Instead of picking the estimate where the fitted curve intercepts a certain value we take the estimate whenever the derivative reaches a predetermined value. This approach gives us the possibility to increase the estimates in general as well as avoiding making too large estimates when the faults all are easy to find. In order to produce a high estimate the fitted curve must level away early to reach far to the right before intercepting the stop level. However if reacting on the derivative the estimation level would have been reached before the curve became too flat. In this case, the estimation rule becomes dependent on the form of the fitted exponential function and the data used to determine the derivative parameter. This modification of DPM is denoted DPM_{der}. The resemblance between the DPM_{mod} estimator and the DPM_{der} estimator implies that the latter also is dependent on the number of inspectors.

2.3. Experimental set-up

Four different estimators are used in this study. First the two modified versions of the DPM, DPM_{mod} and DPM_{der} then the original DPM in order to determine if there have been any improvements. To compare the performance of the modifications to other

estimators, i.e. capture-recapture estimators, the Jackknife estimator, denoted MhJK[1] are used. MhJK is chosen because it has been shown as being the best estimator in a number of studies, for example, [Briand97, Miller99, Petersson99b].

30 data sets from inspections have been used, see Table 1. Of these 20, denoted fit, are used to find suitable parameters for DPM_{mod} and DPM_{der} and 10 are used for evaluation, denoted test. The data sets are the same as used in [Petersson99b]. A new randomisation of which data sets to use as fit and as test is made. For each data set all possible combinations of the participating inspectors are created. These inspections are denoted virtual inspections.

As mentioned in Section 2.2, the parameters are dependent on the number of inspectors. To investigate this, but also to determine the estimators' behaviour when different number of reviewers participate, we study virtual inspections with three, four and five inspectors respectively. In some of the generated virtual inspections some cases appear where no overlap exists between the inspectors. With no overlap the three DPM variants fail to estimate. These cases are relatively few and are excluded from the study altogether to enable a fair comparison against the MhJK.

Table 1: Data sets.

No.	Name	No. of reviewers	Used for	Ref.	No.	Name	No. of reviewers	Used for	Ref.
1	AdhAtmJun	8	Fit	[Freimut97]	16	PbrNANov	6	Test	[Freimut97]
2	AdhAtmNov	6	Fit	[Freimut97]	17	PbrNBJun	7	Fit	[Freimut97]
3	AdhPgJun	6	Fit	[Freimut97]	18	PbrNBNov	6	Test	[Freimut97]
4	AdhPgNov	6	Test	[Freimut97]	19	PbrPgJun	8	Test	[Freimut97]
5	ChklATM	6	Fit	Unpublished[a]	20	PbrPgNov	6	Fit	[Freimut97]
6	EngDMod	7 (22)	Test	[Wohlin95]	21	PbrStatA	8	Fit	[Freimut97]
7	NasaAJun	7	Fit	[Freimut97]	22	PbrStatB	7	Test	[Freimut97]
8	NasaANov	6	Fit	[Freimut97]	23	PbrTextA	8	Test	[Freimut97]
9	NasaBJun	6	Test	[Freimut97]	24	PbrTextB	7	Fit	[Freimut97]
10	NasaBNov	6	Fit	[Freimut97]	25	PbrZinsA	8	Fit	[Freimut97]
11	PBRAtmMod	7 (15)	Fit	[Regnell99]	26	PbrZinsB	7	Test	[Freimut97]
12	PBRPgMod	7 (15)	Fit	[Regnell99]	27	Cdata3A	5	Fit	[Runeson98]
13	PbrAtmJun	6	Fit	[Freimut97]	28	Cdata4A	5	Fit	[Runeson98]
14	PbrAtmNov	6	Fit	[Freimut97]	29	Cdata5A	5	Fit	[Runeson98]
15	PbrNAJun	6	Test	[Freimut97]	30	Cdata6A	5	Fit	[Runeson98]

a. Used in [Regnell99] though the data set is not published.

A threat to this study is the fact that the data sets are not totally independent of each other. In some experiments, the same documents have been inspected. Though in each case it has been different people performing the inspection. There are also cases where the same people have participated in more than one of the experiments. These dependencies could lead to lower deviations in the estimation results. The creation of the virtual reviews also adds to the dependence. By utilising all the combinations of the reviewers the same reviewer will be part of many virtual reviews. However, since the study is made by comparing different estimators all dealing with the same data sets, all of the estimators should have the same benefits and it should still be possible to compare their relative behaviour. If using the methods in an industrial setting the situation of one person participating in many inspections should be common.

1. To calculate MhJK the algorithms from the program CAPTURE [Rexstad91], version of 16th May 1995, have been used.

2.4. Results

First, the parameters for the two modified methods DPM_{mod} and DPM_{der} are determined. To do this, all virtual inspections for the fit data sets are created. Then the decreasing exponential curve is fitted for each virtual inspection. In each case, the y-value and the derivative of the curve at the place where x is equal to the true number of defects are recorded. This is made separately for three, four and five reviewers. The parameters are then picked as the mean of these collected values. The resulting parameters are shown in Table 2.

Table 2: Parameters for DPM_{mod} and DPM_{der}

	DPM_{mod}	DPM_{der}
3	0.274	-0.0260
4	0.294	-0.0314
5	0.335	-0.0381

With the parameters determined, all virtual inspections are created and estimations are done for the ten test data sets.

To compare and combine the results, the relative error of each estimate is calculated. The relative error (RE) is defined as:

$$RE = \frac{\text{Estimated number of defects} - \text{Actual number of defects}}{\text{Actual number of defects}}$$

A boxplot of the resulting relative errors for the case of 4 inspectors is shown in Figure 2. The bottom and top border of the boxes in the boxplots show the 25 and 75% quartiles and the line inside shows the 50% quartile. The whiskers are extended 1.5 times the inter-quartile range from the 50% quartile level. The data points outside the whisker range are marked with a '+' as outliers. The mean values of the relative error can be seen in Table 3.

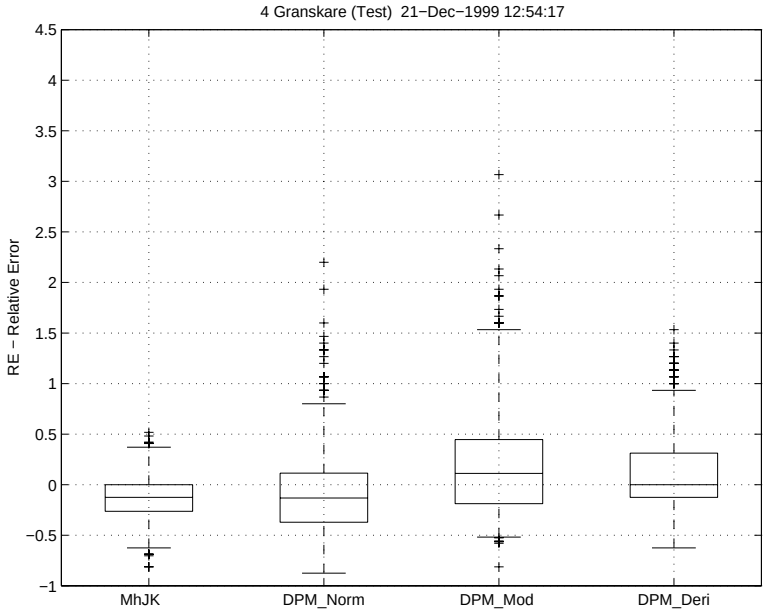

Figure 2. Results for 4 reviewers

Table 3: Mean value of Relative Error

	Mean
MhJK	
DPM	
DPM_{der}	
DPM_{mod}	

Note that the 50% quartile level of the boxplot represents the median and not the mean value. The boxplots for three and five inspectors respectively show similar results.

2.5. Analysis

An examination of the boxplots leads to three initial conclusions.

1. The DPM_{der} has improved the deviation and slightly improved the bias of the original DPM.
2. The estimator MhJK has the least deviation and would probably be considered as the best estimator though with a negative bias since we obtain about 75% underestimations.
3. DPM_{mod} does not behave well, however according to the expectation the bias is increased and deviation has increased compared to the original DPM.

These results are valid when considering all of the test data sets at the same time since the boxplot shows the union of all results. To investigate further, the mean and standard deviation in the four-inspector case are calculated for each data set and presented in Table 4.

Table 4: Mean and Standard Deviations of Relative Error

	Mean	Std Dev.		Mean	Std Dev
	1			6	
DPM	-0.491	0.081	DPM	0.684	0.530
DPM_{mod}	-0.407	0.104	DPM_{mod}	1.143	0.662
DPM_{der}	-0.323	0.076	DPM_{der}	0.813	0.304
MhJK	-0.195	0.294	MhJK	0.006	0.086
	2			7	
DPM	-0.097	0.092	DPM	-0.068	0.141
DPM_{mod}	0.132	0.118	DPM_{mod}	0.193	0.187
DPM_{der}	-0.117	0.040	DPM_{der}	0.018	0.073
MhJK	-0.053	0.149	MhJK	-0.003	0.213
	3			8	
DPM	0.627	0.477	DPM	-0.376	0.145
DPM_{mod}	1.027	0.606	DPM_{mod}	-0.221	0.213
DPM_{der}	0.827	0.317	DPM_{der}	-0.104	0.151
MhJK	-0.071	0.069	MhJK	-0.048	0.265
	4			9	
DPM	-0.159	0.137	DPM	-0.483	0.152
DPM_{mod}	0.087	0.186	DPM_{mod}	-0.306	0.204
DPM_{der}	-0.074	0.057	DPM_{der}	-0.141	0.165
MhJK	-0.060	0.207	MhJK	-0.491	0.180
	5			10	
DPM	0.644	0.285	DPM	0.004	0.181
DPM_{mod}	1.071	0.372	DPM_{mod}	0.266	0.231
DPM_{der}	0.871	0.175	DPM_{der}	0.341	0.156
MhJK	0.089	0.048	MhJK	-0.195	0.060

When studying Table 4, conclusions are harder to draw. MhJK can still be seen as the best estimator as its bias (mean) is ranked best in eight of the ten cases even if its deviation is ranked last five times. When studying each data set, the DPM_{mod} must still be considered as behaving worst of the four estimators. Six times its mean is ranked last and five times its deviation. Only one time the DPM_{mod}'s mean or standard deviation is ranked as high as second, and it is never ranked as the best.

It is more difficult to determine which of DPM and DPM_{der} that performs best. When comparing the rank of the means of these two, it comes to a draw (5-5). For the standard deviation however the count is 8-2 in DPM_{der}'s favour. The deviation of DPM_{der} is ranked either first or second for all ten data sets. This even outranks MhJK's performance.

The results of the ranks for three and five reviewers are similar though with a few minor changes.

As DPM_{mod} behaves worst of the modifications only DPM_{der} is considered in the next section.

3. Calibration of the Estimation Rule

At first, the modification of the DPM was intended as a general improvement with fixed values on the estimation parameter, although with different values depending on

the number of inspectors. However, instead of seeing the parameter as fixed it can be seen as an opportunity for a continuous calibration. The parameter could be calibrated using historical data so that within a company or department the parameter would adapt to the history and it would evolve as new inspections are performed. In this case, the procedure of calculating the parameter based on such a broad range of data sets as the 20 used in this study, is improper.

Two scatter plots of the values for DPM_{der} where the fitted curve intercepts y = "true number of fault" are shown in Figure 3. The left plot shows the plot for the fit data sets and the right plot for the test data sets. As described in Section 2.4, the parameter for DPM_{der} was found to be -0.0314, which is the mean of the values in the left plot in Figure 3. Based on the right plot, it is easy to identify the data sets where the DPM_{der} fails the most.

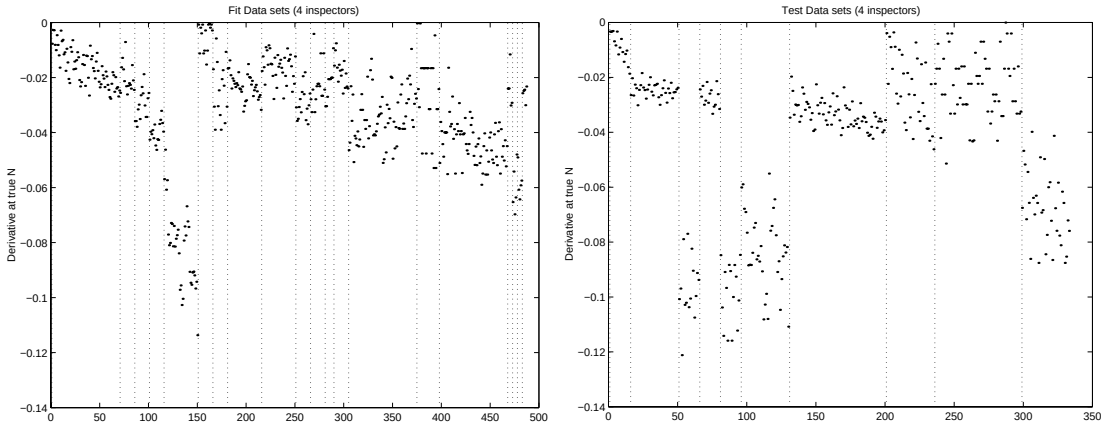

Figure 3. Scatter plots for the values when the fitted curve intercepts y = "true number of faults".

If all data sets had been taken from the same company or the same department the similarities between the inspections and there by the data sets would have been greater. The inspection technique may have been more similar as well as the types of documents. The range of people that could participate as inspectors would also be limited. Such a convergent situation would present a better basis for a calibration relying on historical data.

In order to investigate whether similarities between data sets may help improve, the data sets have been classified with respect to three aspects:

- Environment (Students, Faculty, NASA, Professionals other than NASA)
- Document type (Req., Code etc.).
- Inspection technique (Ad hoc, Checklist or Perspective-Based Reading (PBR) [Basili96])

Table 5 shows the thirty data sets with the mean and standard deviation of estimation parameter of DPM_{der} together with the classification. The data sets have been sorted in increasing order according to the mean value. The term artificial has been used when a document has been either developed or adapted specifically for a controlled experi-

ment. In other words, the document does not come directly from a real software project.

Table 5: Classification of Data Sets

No.	Name	Mean	Std Dev.	Fit/Test	Env.	Doc. Type	Insp. Tech.
1	NasaANov	-0,0050	0,0051	F	NASA	Req	Adh
2	AdhPgNov	-0,0091	0,0049	T	NASA	Fake Req	Adh
3	AdhAtmJun	-0,0161	0,0066	F	NASA	Fake Req	Adh
4	PBRPgMod2	-0,0163	0,0052	F	Stud.	Fake Req	PBR
5	PbrPgNov	-0,0180	0,0052	F	NASA	Fake Req	PBR
6	AdhAtmNov	-0,0201	0,0053	F	NASA	Fake Req	Adh
7	PbrNBNov	-0,0221	0,0065	F	NASA	Req	PBR
8	PbrNANov	-0,0227	0,0080	F	NASA	Req	PBR
9	PbrTextA	-0,0230	0,0116	T	Prof.	Code	PBR
10	PBRAtmMod2	-0,0231	0,0045	F	Stud.	Fake Req	PBR
11	Cdata3A	-0,0238	0,0074	F	Prof./Stud.	Code	Chkl
12	PbrStatB	-0,0243	0,0127	T	Prof.	Code	PBR
13	EngDMod2	-0,0246	0,0027	T	Prof./Stud.	Textual	Adh
14	NasaBNov	-0,0249	0,0104	F	NASA	Req	Adh
15	Cdata6A	-0,0257	0,0026	F	Prof/Stud	Code	Chkl
16	PbrTextB	-0,0257	0,0158	F	Prof.	Code	PBR
17	PbrAtmNov	-0,0272	0,0037	T	NASA	Fake Req	PBR
18	PbrAtmJun	-0,0294	0,0057	F	NASA	Fake Req	PBR
19	AdhPgJun	-0,0302	0,0055	F	NASA	Fake Req	Adh
20	PbrPgJun	-0,0334	0,0049	T	NASA	Fake Req	PBR
21	PbrStatA	-0,0342	0,0094	F	Prof.	Code	PBR
22	ChklATM	-0,0398	0,0038	F	Stud./Faculty	Fake Req	Chkl
23	PbrZinsA	-0,0436	0,0077	F	Prof.	Code	PBR
24	Cdata5A	-0,0581	0,0057	F	Prof./Stud.	Code	Chkl
25	Cdata4A	-0,0602	0,0089	F	Prof./Stud.	Code	Chkl
26	PbrZinsB	-0,0674	0,0133	T	Prof.	Code	PBR
27	NasaAJun	-0,0816	0,0143	F	NASA	Req	Adh
28	PbrNBJun	-0,0832	0,0141	T	NASA	Req	PBR
29	NasaBJun	-0,0966	0,0115	T	NASA	Req	Adh
30	PbrNAJun	-0,0987	0,0114	T	NASA	Req	PBR

When studying Table 5 no obvious pattern of the characteristics can be identified. For example, for the four data sets with the largest bias it is not possible to find a common denominator for these four which differentiate them form all other data sets. All four are taken from a study made at NASA but other data sets from the same study are present without having such large bias, e.g. number 3 and 18-20. The four latter are, however not based on real requirements specifications. On the other hand, it is hard to see any pattern if just looking at real specifications. In conclusion it must be noted that such a simple classification as this fails to capture the underlying differences that may explain the behaviour.

In order to evaluate if the calibration of DPM_{der} would benefit from a homogenous setting there is a need for such data sets. Among the thirty data sets there are only four in which the same technique, type of document and group of people have been used. These data sets are the C-data sets collected from [Runeson98]. In this study Runeson et al. performed inspections of five c-code programs with eight different inspectors. The inspectors were randomly assigned into groups to inspect each code document[1]. The code documents were taken from assignments in Watts Humphrey's PSP course [Humphrey95]. To ensure authenticity the code was saved after coding before any defects were removed. Defects were then later found in inspections, compile and test-

ing and these faults are viewed as the “correct” answer in the experiment. For further details concerning the experiment please refer to [Runeson98].

As in the evaluation with the thirty data sets, the four C-data sets were divided into fit and test. The first two, 3A and 4A, were used as fit to calculate the estimation parameter. This parameter, found to be -0.0420, was then used when calculating the estimates for the virtual inspections of data set 5A and 6A. A scatter plot of the relative errors of the estimations can be seen in Figure 4. Table 6 shows the mean and standard deviation.

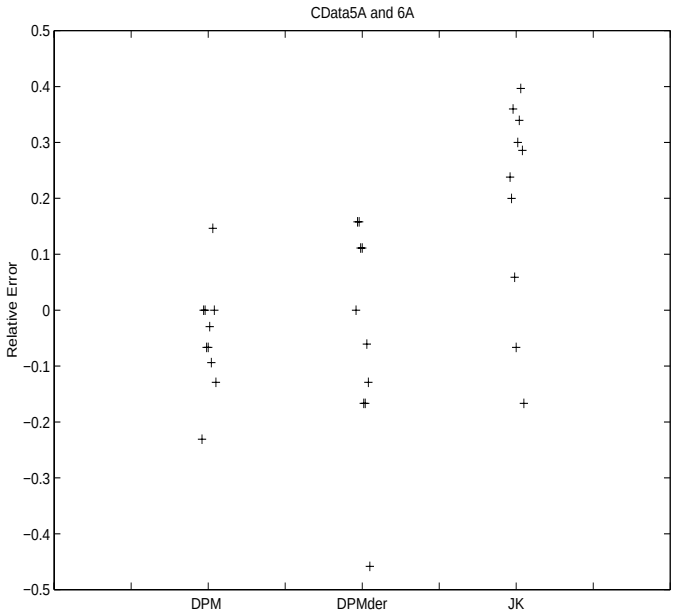

Figure 4. A scatter plot of the relative errors.

Table 6: Mean and deviation for C Data Run

	Mean	Std Dev
DPM_{der}	-0.015	0.168
MhJK	0.298	0.270
DPM	-0.037	0.094

The plot and the values from Table 6 show that DPM_{der} is better than MhJK but at the same time DPM performs best. The mean of DPM is ranked best and the deviation is only half of MhJK’s or DPM_{der}’s. DPM_{der}’s mean is almost as good as DPM and only one single outlier spoils its deviation.

This short evaluation has few data points so it is hard draw any general conclusions from it. Though the evaluation may imply that the calibration of the estimation parameter in DPM_{der} is feasible compared to MhJK. Further studies in stable environments are however needed to evaluate the usefulness of a company specific parameter in the DPM_{der}.

1. Only 4 of the inspection data sets were used here. The last one had too few reviewers compared to the others.

4. Summary and Conclusions

The focus of this paper was to investigate and evaluate two modifications of the DPM. The first modification was unsuccessful since the positive change in mean value was too small compared to the negative change in deviation. The other modification, when the derivative was used as the estimation parameter, did not show these negative results, instead the deviation was improved. However, the improvement of the bias level was not apparent.

Instead of seeing the modification as a permanent change, the parameter could be seen as calibrated from historical data. The C-Data sets available in this study gave no clear indication but in order to fully investigate this more studies are needed. An advantage with this calibration is that the procedure with a continuous calibration from historical data should fit an industrial setting very well.

Even if the historical calibration could not be fully investigated the outcome of this study is that MhJK must be considered as the best estimator. The main advantage is fairly good estimates without being dependent on the history. MhJK does, however, make a lot of underestimations and could gain on being calibrated too. A calibration/ bias correction of MhJK was performed as a part of the study in [Petersson99b]. It was shown that MhJK's estimates could be raised without increasing the deviation too much, but the advantage of MhJK of not being dependent on historical data is then lost.

This advantage is important since it takes some effort to collect the necessary data in order to calibrate an estimator. All faults have to be carefully recorded and classified of where they where injected. It is then possible to do a backward calculation from the number of total faults in the product to determine the number of faults present at the time of the inspection. The accuracy of this calculation will not be good until the product spent a long time in the maintenance phase allowing for all faults to be found.

The calibration may however be performed in other ways. A controlled experiment at the company could be executed. The participating people, the document types and the inspection technique would be the same as used normally, but the documents would have a number of seeded faults. The results from these trial runs could then be used to calculate the estimation parameters to be used while waiting for better ways of calibrating to become available.

In order to improve an estimator that underestimates, it is desirable to increase the estimate without increasing the deviation. Otherwise, the already high overestimates will increase and become a problem of their own. This leads to a question of what is worst, an underestimation or an overestimation? This is not so easy to decide and it has to be treated on a case by case basis. If the estimate is used as a stopping rule and/or quality stamp for inspections then an underestimation leads us to think that the quality is better than it is, but it takes less time since we decide to not re-inspect. However this decision might affect the future and something else would probably take time in the future. An overestimation, on the other hand, leads to unnecessary work directly if making a re-inspection. So it is at matter of what to prioritise. Should we prioritise time-to-market or quality? This balance depends on so many parameters and may change over time.

Since a couple of studies has been performed, which results in that MhJK performs best, the research of capture-recapture in software engineering should change its focus.

The focus so far has been to evaluate and improve specific estimators. But since MhJK seems to be picked as the best in many of the studies, it would be interesting to focus on how the defect content estimation models should be used and also how accurate estimators the industry needs in order to find capture-recapture useful? Example of this change in focus can be seen in, for example, [El Emam99, Stringfellow99]

As a recommendation for industrial practice, we would like to, based on our findings, recommend using the MhJK estimator in combination with a simple experience based approach. The latter could either mean bias correction as discussed in [Petersson99b] or by using the DPM_{der} estimator after having evaluated it carefully for a specific environment.

Acknowledgement

We would like to thank Thomas Thelin at the Department of Communication Systems, for his valuable comments on this paper. This work was partly funded by The Swedish National Board for Industrial and Technical Development (NUTEK), grant 1K1P-99-6121.

References

[Basili96] V. R. Basili, S. Green, O. Laitenberger, F. Lanubile, F. Shull, S. Sørumgård and M. V. Zelkowitz: “The Empirical Investigation of Perspective-Based Reading”, Empirical Software Engineering: An International Journal, Vol. 1, No. 2, pp. 133-164, 1996.

[Briand97] L. Briand, K. El Emam, B. Freimut and O. Laitenberger: ”Quantitative Evaluation of Capture-Recapture Models to Control Software Inspections”, In Proc. of the 8:th International Symposium on Software Reliability Engineering, pp. 234-244, 1997.

[Briand98] L. Briand, K. El Emam and B. Freimut: “A Comparison and Integration of Capture-Recapture Models and the Detection Profile Method”. In Proc. of the 9:th International Symposium on Software Reliability Engineering, 1998.

[El Emam99] K. El Emam, “???

[Freimut97] B. Freimut, “Capture-Recapture Models to Estimate Software Fault Content”. Diploma Thesis, University of Kaiserslautern, Germany, June 1997.

[Humphrey95] W. S. Humphrey: A Discipline for Software Engineering. Addison-Wesley, USA 1995.

[Miller99] J. Miller, “Estimating the Number of Remaining Defects after Inspection”. In Journal of Software Testing, Verification and Reliability, Vol. 4, No 9, pp. 167-189, 1999.

[Otis78] D. Otis, K. Burnham, G. White and D. Anderson: “Statistical Inference from Capture Data on Closed Animal Populations”. Wildlife Monographs, No. 62, October 1978.

[Petersson99a] H. Petersson, C. Wohlin: “Evaluation of using Capture-Recapture Methods on Software Review Data”. In Proc. of the Third International Workshop on Empirical Assessment and Evaluation in Software Engineering, Keele University, Staffordshire, UK April 12th - 14th 1999.

[Petersson99b] H. Petersson, C. Wohlin: “An Empirical Study of Experience-based Software Defect Content Estimation Methods”, In Proc. of the 10:th International Symposium on Software Reliability Engineering, Boca Raton, Florida, USA, 1-4 November, 1999.

[Regnell99] B. Regnell, P. Runeson, and T. Thelin “Are the Perspectives Really Different? - Further Experimentation on Scenario-Based Reading of Requirements”, Technical Report CODEN: LUTEDX(TETS-7172) / 1-38 / 1999 & local 4, Dept. of Communication Systems, Lund University, 1999.

[Rexstad91] E. Rexstad and K. P. Burnham, User’s guide for interactive program CAPTURE, Colorado Cooperative Fish and Wildlife Research Unit, Colorado State University, Fort Collins, CO 80523, USA, 1991.

[Runeson98] P. Runeson and C. Wohlin: “An Experimental Evaluation of an Experience-Based Capture-Recapture Method in Software Code Inspections”, Empirical Software Engineering: An International Journal, Vol. 3, No. 4, pp. 381-406, 1998.

[Stringfellow99] C. Stringfellow, A. von Mayrhauser, C. Wohlin, H. Petersson: “Estimating the Number of Components with Defects Post-Release that Showed No Defects in Testing”, Submitted to Journal of Software Testing, Verification and Reliability, 1999.

[Wohlin95] C. Wohlin, P. Runeson and J. Brantestam: “An Experimental Evaluation of Capture-Recapture in Software Inspections”, In Journal of Software Testing, Verification and Reliability, No. 4, pp. 213-232, 1995.

[Wohlin98] C. Wohlin and P. Runeson: “Defect Content Estimations from Review Data”, In Proc. of the 20th International Conference on Software Engineering. pp. 400-409, 1998.