Slope of a Line, More Than Just Math – D.A.R.E to Add it Up.

Providing predictability and project management discipline to software testing and defect management.

Why can a lawyer tell you how long it will take to prepare a legal document or an auto body repair shop tell you it will take 1.5 hours to replace a bumper on your BMW?

The answer is hidden in years of repeatability, data collection and math!

So why in Information Technology, a field built on math, do we all too often find it so hard to forecast, predict and estimate? To build upon one of our favorite BusinessWeek cover stories "Why Math Will Rock Your World" – looking for patterns and applying mathematical techniques to problem solving can be a very powerful tool. By applying some practical discipline, easily available tools and math patterns, you can add fact-based value to your next software development project.

Due to the many moving parts in an IT organization, there exists a huge challenge in accurately predicting testing costs and timelines. Mark Robinson, Assistant Vice President of Information Systems Administration Unit for the Ohio Casualty Group states, "Developing software systems for the insurance industry is extremely complex. A system that writes personal line policies, for example, must not only meet our own business requirements but must also support every relevant state rule and federal regulation. There are a million combinations. When you have a system that complex with that many variables, the testing effort is monumental [1] ."

Couple the above with the massive quantity of software (over 8 billion lines of software code developed and or modified each year [2]), and the complexity of teams developing software globally, quality is a daunting issue! We are going to D.A.R.E to help.

---

[1] http://www-01.ibm.com/software/success/cssdb.nsf/CS/WJBN-6PLHJS?OpenDocument&Site=software
[2] http://www.cmcrossroads.com/content/view/9082/135/

The D.A.R.E Framework

D.A.R.E is simply an acronym for **D**ata Collection, **A**nalysis, **R**egression and **E**xcel, a regression-based framework that uses historical data to confidently predict future events. It is our belief that with some good data collection, analysis, mathematic regression techniques and Microsoft ® Excel (or any spreadsheet tool), you can amplify the power of predictability.

Much like the lawyer or auto repair shop technician, the next time your project manager, boss or client asks you how many developers are needed to support the test phase or how many defects are expected in this release, you can give them a response grounded in fact and sound thinking.

The "Right" Project for D.A.R.E

As powerful as D.A.R.E is, not all projects are candidates for a regression-based framework. Before deciding whether to adopt this approach, your project should have at least the following characteristics:

1. The development lifecycle needs to be iterative – In other words, the project needs to have multiple build and test phases to provide a baseline from which to predict. For example, if the project has only two iterations of development and testing, then enough data points do not exist to develop a workable model. We'll talk about how many data points is "enough" in our "Analysis" section.

2. Iterations must be mutually exclusive – If enough iterations exist, it is important to ensure each one is self-contained. For example, if you are trying to use D.A.R.E to predict the total number of defects in a phase of development, the defects must reside in only one phase to avoid "double-counting".

3. Accurate data must be available – Remember GIGO: Garbage In – Garbage Out! Many quality tools are available to capture data as part of development and testing efforts; two popular tools are IBM's Rational ClearQuest and HP's Mercury Quality Center. These tools

are configurable and can be used to collect any number of data elements. They also help in the generation of reports and metrics.

Data Collection:

Data collection is the first, and arguably the most important part of the D.A.R.E framework. A major prerequisite in being able to identify trends and generate accurate estimates is through the collection of high quality data. With the quality management tools that are available and widely used in software development, the task of consistently gathering high quality data becomes a relatively easy one.

You can ensure that your data is reliable by following two simple guiding principles. The first point addresses the process by which data is collected and the second addresses the importance of consistently following those processes.

Process of Data Collection

While a quality management tool can assist greatly in the collection of data, processes must be built around the tool to ensure it is used as intended. A good example of this is the closure status of a defect. Work can be concluded on a defect in a variety of ways. A defect may end up in duplicate, resolved, rejected, closed, or any number of statuses. The important point is to make this decision explicit in a process and make sure that process is documented and communicated to all members of the project team.

Consistency of Execution

After the process is established, it is very important to ensure that all groups are consistent in following that process. In some projects, it becomes very difficult to determine the root cause of a defect due to a lack of consistency in the defect management process. In this scenario, defect analysis can only be performed on the portion of the defects owned by groups that followed the same process. This lack of consistency can greatly hinder the ability to create and utilize predictive modeling. Therefore be consistent!

Components of Data Collection (Dependent and Independent Variables)

Once you have consistent processes in place for data collection, ask yourself what is it you are trying to predict? Whatever you are ultimately trying to solve for is called the dependent variable. The variables that influence your dependent variable are called the independent variables. For example, the dependent variable can be the number of defects or defect severity, and independent variables can be number of lines of code or a measure of complexity level. A key point is that multiple independent variables can be used to predict a single dependent variable. Both sets of data must be collected to produce a regression model.

Below are a few points to think about when creating your data collection strategy:

1. Judge the complexity of each iteration – It's a fair statement that developing and testing a missile-guidance system is probably more complex than a basic calculator. Therefore, it is important for the project manger, with the support of the project architect and business analyst, to invest time devising a framework for determining the complexity of each iteration. For example, a Property and Causality insurance project manager can determine the complexity of a state by examining the state filings with the department of insurance. So states like NJ and CA would be complex and IL and IN would be considered simple[3]. We'll revisit the insurance industry again later in our "Excel" section.

   While a framework for determining complexity is not always clear, it can be derived through a logical thought process. One suggestion is to start by using a simple number scale ranging from one to twenty to judge the complexity of each project iteration. Next, take a close look at the measurable data elements that contribute to the project's complexity. Examples can include the number of requirements, application integration points, or the size of the release. Size can be determined by the number of developers, lines of code, or any other measurable data point.

   As data and results are collected, trends, constants, and themes will begin to emerge helping to indicate data elements (independent variables) that will influence your dependent variable.

[3] Based on IBM Research (GBS Insurance Practice - 2009).

2. Identify independent variables as early as possible – Adding or changing independent variables can change the model in sometimes surprising ways and may require significant time to go back and gather or re-verify your data.

3. Iterations must use the same independent variables – Each iteration of the development lifecycle must harvest the same independent variables to include in the data collection process. Due to point 2 above, error on the side of collecting more data than may be required rather than not enough.

4. Avoid too much positive correlation – To develop a model with strong predictive power, avoid only adding variables that correlate strongly with one another. If possible, include independent variables that are negatively correlated. This may contribute to creating a more accurate regression model. Additional details are provided below in the correlation matrices section under "Analysis".

5. Only add independent variables that have significant predictive power – Adding multiple independent variables to your model are only valuable if they have "predictive power". A good way to test whether a potential independent variable has this characteristic is by using Scatterplots and Correlation Matrices, which are all available in Microsoft Excel. These techniques will be discussed further in the "Analysis" portion of the framework.

Analysis:

As the data begins coming in during data collection, it is important to constantly monitor how your model is developing using simple tools within Microsoft Excel or any other spreadsheet tool. As a result of your analysis, you can begin to select those independent variables that have the greatest potential to have significant predictive power. In this section, we'll look at some of the tools available and provide examples to assist you.

Scatterplots

One of the most useful tools available, Scatterplots are a great way to determine how well a single independent variable will predict the dependent variable.

How to interpret a Scatterplot – In the following Insurance Industry case study, the project team is trying to determine the total number of defects that will be found during a fixed testing schedule. Therefore the dependent variable in this example is the number of defects. The project team suspects the number of testing days greatly impacts the number of defects found and wants to test this theory. The number of business days of testing will be the independent variable.

Two iterations of the testing phases are shown below. The x-axis is the number of business days of testing and along the y-axis is the total number of defects. By applying a line-of-best-fit, the project team can identify if there is a strong relationship between the number of days of testing and the total number of defects.

Taken separately, each graph shows a fairly tight line-of-best-fit as well as a high R^2 (See Regression section for more information on the R^2 statistic). The slope of both lines remains relatively constant throughout the course of testing with very high R^2 of .93 and .97 respectively.

Figure 1 -Release 2 (R1 Not Graphed, see Raw Data for complete data set):

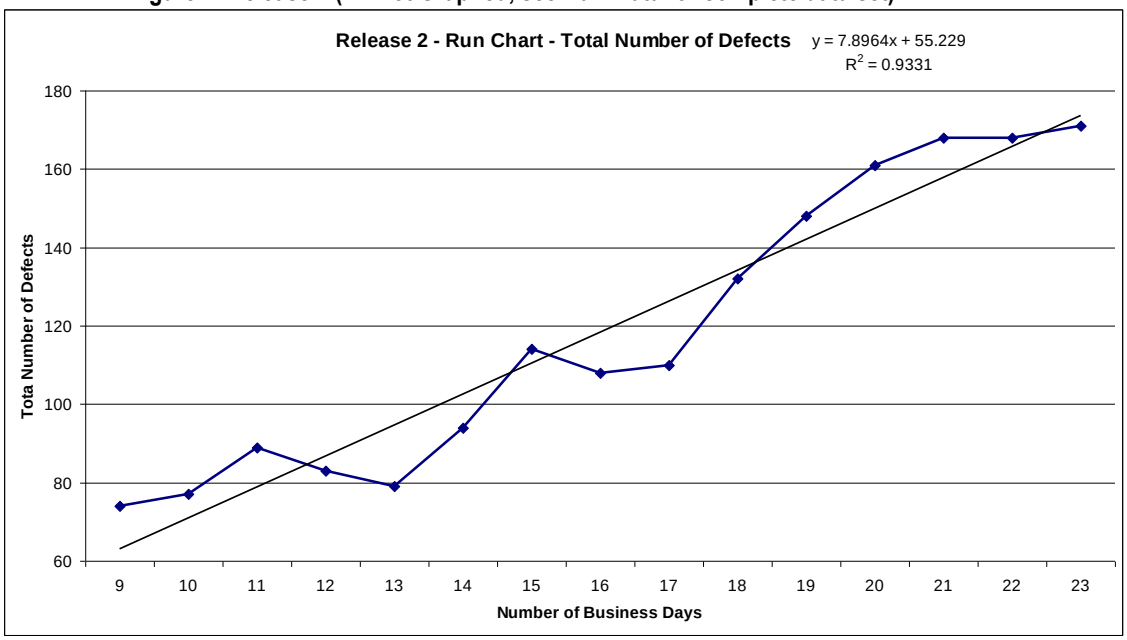
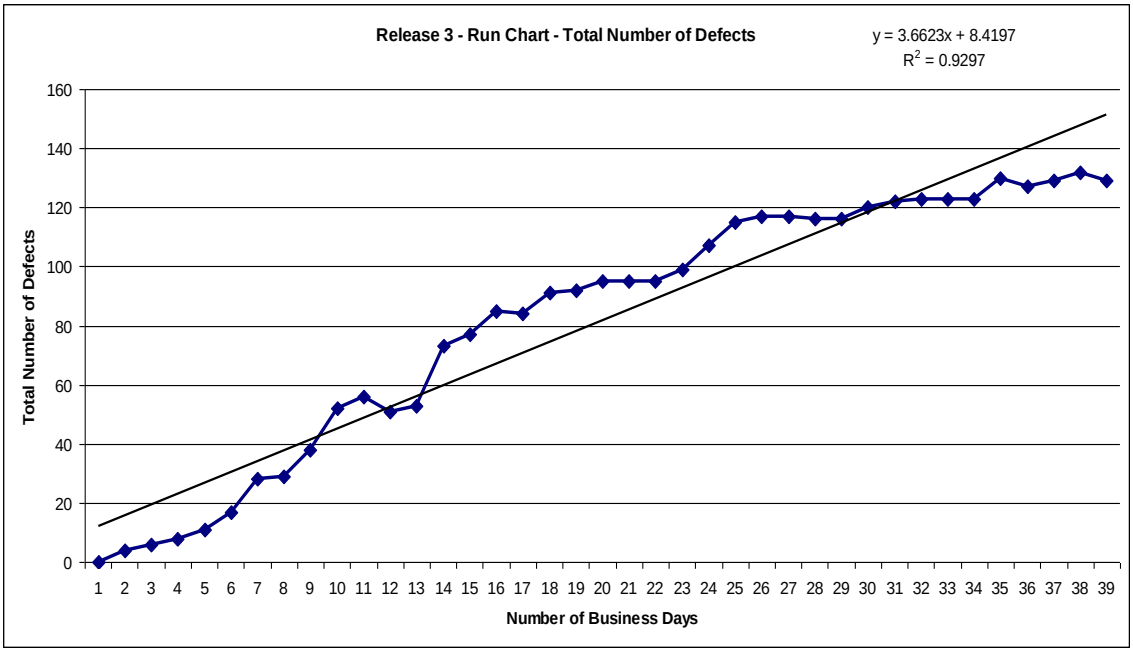
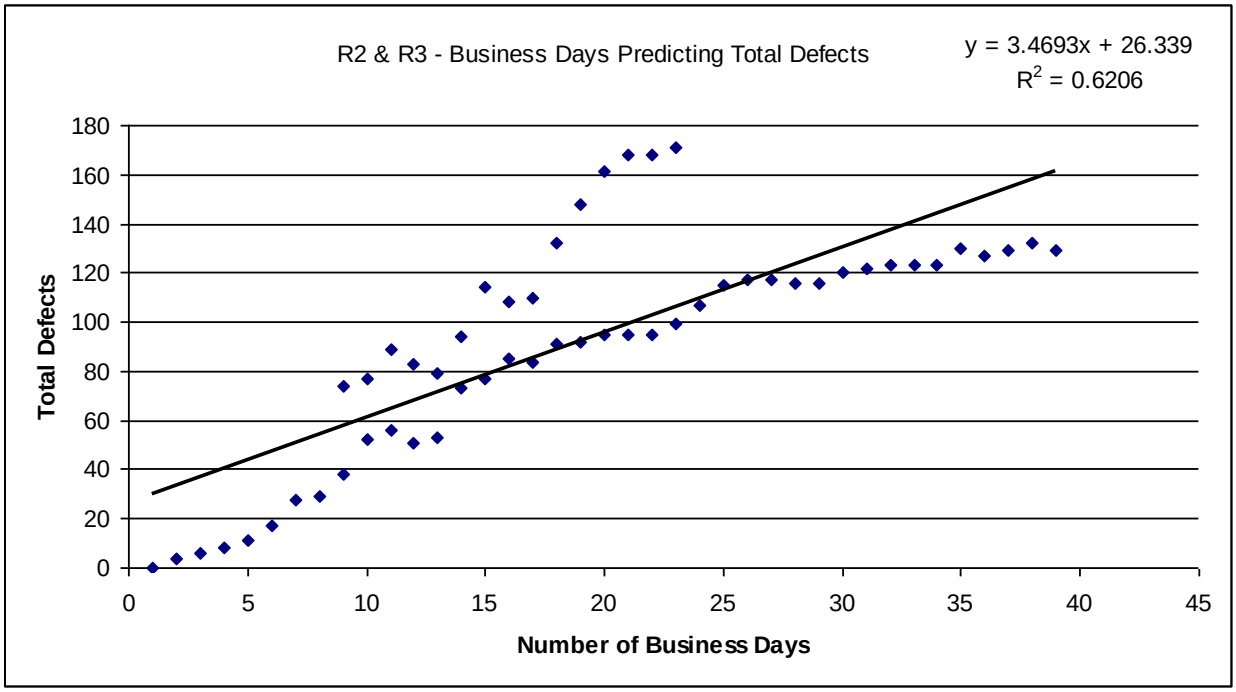


Figure 2 - Release 3:



Unfortunately, the total number of testing days alone will not be enough. As seen in the chart below, when both release three and four are plotted together, the slopes of both lines differ significantly, resulting in a much lower R^2 of .62 and a larger deviation between the data points and the line-of-best-fit. Therefore, some OTHER variable must play a role in determining the total number of defects.

Figure 3 - Release 2 & 3:



In summary, Scatterplots are a great tool to begin understanding what independent variables impact your dependent variable and give you insight into whether or not additional variables are needed to make your model more powerful. In our example above, the number of days of testing is helpful, but additional variables must be examined.

Correlation Matrices

Once Scatterplots have helped identify individual independent variables with predictive power, it is time to compare them to each other. Correlation matrices can be used to do this. They will determine how well each variable, both independent and dependent, correlate inside one table. This tool can be used to eliminate any variables with small predictive power and identify those variables that can ultimately be included in the final regression model.

How to interpret a correlation matrix

Each value in the correlation matrix has a value that can range from -1 to +1. Two values that are negatively correlated (a correlation value approaching -1) means that as one value increases, the other decreases. Two values that are positively correlated (a correlation value approaching +1) means that as one value increases, the other increases. As mentioned in the data collection phase, it is good to include independent variables that have both strong positive and negative correlations with the dependent variable.

Using another case study from the insurance industry as an example, we generated a correlation matrix using excel with three independent variables and a dependent variable. Variable descriptions and the correlation matrix are shown below:

Independent Variables

1. Total Number of Days in Test (A) – This is the total number of business days allocated to the testing effort.
2. Total Number of States in Release (B) – Each release included a varying number of states (e.g. – New York or Florida).
3. Complexity Scores (C) – This is an objective complexity score allocated to each release based on objective criteria discussed in our "Data Collection" section.

Dependent Variable

1. Total Number of Defects (D) – This is the total number of defects ultimately created and resolved over the duration of the testing effort.

Correlation Matrix	Total # of Business Days in Test (A)	Total # of States in Release (B)	Complexity Scores (C)	Total # of Defects (D)
(A) Total # of Business Days in Test	1.00			
(B) Total # of States in Release	.66	1.00		
(C) Complexity Scores	-0.67	-0.03	1.00	
(D) Total # of Defects	-0.76	-0.10	0.87	1.00

Based on this matrix, there is a strong correlation between the following independent variables and the dependent variable:

1. (A) Total # of Business Days in Test (-.76)
2. (C) Complexity Scores (.87)

Correlations at or near zero indicate a weak relationship with the dependent variable and can be removed from the regression model. For example, there is a weak correlation between "Total Number of States" and "Total Number of Defects". As a result, this variable can be excluded, leaving two independent variables remaining.

Correlation matrices are powerful, but it should be noted that they do not imply causality. In other words, just because there is a strong positive correlation between "Complexity Scores" and "Total # of Defects" you should not assume that the higher complexity score CAUSED the total # of defects to increase. There could in fact be a third variable causing both of them to increase.

Regression:

Up to this point, we've discussed how the appropriate collection and selection of data will create a predictive model. In this section, we will explain how the model itself works in greater detail. The goal of the regression model is to analyze the relationship between our dependent variable and the identified set of independent variables to guide the decisions of the project manager.

Using our insurance industry case study again as an example, the equation for the multiple regression line is just an extension of the regular linear equation y = mx + b (slope of a line):

$$\text{(Total Number of Defects) } Y = m_1X_1 + m_2X_2 + m_3X_3 + m_4X_4\ldots\ldots + b$$

The coefficients in this equation are m_1, m_2, m_3, m_4…m_n. With all other variables being equal, the coefficients in a multiple linear regression represent the change in the dependent variable on the y-axis associated with a unit change in a given independent variable on the x-axis.

As will be shown in the "Excel" section of the framework, the regression model will derive these coefficient values for you in order to create your predictive linear equation. Once this equation is created, you are then able to enter your independent variables into the equation in order to come up with your estimate. Your confidence in this estimate will be based on a number of factors described below.

Interpreting a Regression Model

There are many statistical values that are generated as part of a regression model. The following descriptions will guide you in what to look for when judging whether a model is statistically significant, meaning whether the results of the regression model will be able to guide the decision making process. When a model becomes statistically significant, then a baseline set of data points exist from which the project manager can confidently predict.

1. R^2 – This statistic will reflect the percentage of variation explained by the model. The R^2 can vary between 0 and 1. The closer it is to 1, the more variation that is explained by the model, which is a very good thing.
2. F-Statistic – Linked to determining statistical significance, this is the ratio of explained variation to unexplained variation for the regression equation. In other words, the F-Statistic is testing the strength of the relationship between the independent and dependent variables. Since this metric has no upper bound, the higher this number, the better. Also, examine the significance of the F-statistic by comparing it to your confidence interval. If "Significance F" is less than 1-Confidence Interval, the F-statistic ratio is valid.
3. Standard Error – Approximately two times the standard error (exactly 1.96) will give you the spread of your regression model. For example, if the number of defects predicted is 50, but the standard error is 5, then the spread is approximately 10, meaning anywhere between 60 and 40 defects may actual appear in the iteration 95% of the time (assuming a 95% confidence interval). Therefore, the lower this number, the better.
4. P-Value – Also linked to statistical significance, this value indicates what hasn't been explained by the model. As long as 1 – p value is greater than your confidence interval that is acceptable. This applies to the variables in your model, both independent and dependent. If some variables are statistically significant and some are not, then additional steps may need to be taken to either re-evaluate the model itself or wait for more data before making predictions confidently.
5. Confidence Interval – Closely tied to the p-value, the confidence interval is the threshold for determining the significance of the regression model. Generally speaking, most confidence

intervals are set around 90%, meaning that you are 90% "confident" that a random sampling from the same population would yield a mean within approximately two standard deviations of your current mean. Due to the limited amount of data that will most likely accompany most models described in this paper, a confidence interval of 90% is appropriate.

Limitations of the Regression Model

1. Lack of Data – To create a model with any real predictive power, as many iterations as possible as well as high quality data must be incorporated into the model to achieve statistical significance. However, until that point has been reached (see metrics above in "Interpreting a Regression Model"), managers can effectively use the existing data points to create meaningful estimates using already discussed techniques, such as Scatterplots.
2. Prediction outside the range of data – There is only a specific range that can be predicted accurately with the regression model. Data points that exist outside the range have the potential to skew the model and may ultimately become outliers.
3. Statistical Significance but not Managerial Significance – It may be possible to have statistical significance but not much use managerially. The standard error of your model might be too large to be of much use. For example, if the number of defects predicted is 500, but the standard error is 200, then the spread is approximately 400. Therefore, you could say you are 95% confident that the average number of defects will fall between 900 and 100 defects. Unfortunately, this spread is so big that it may not help the project manager make a decision regarding a schedule or estimate.

Excel:

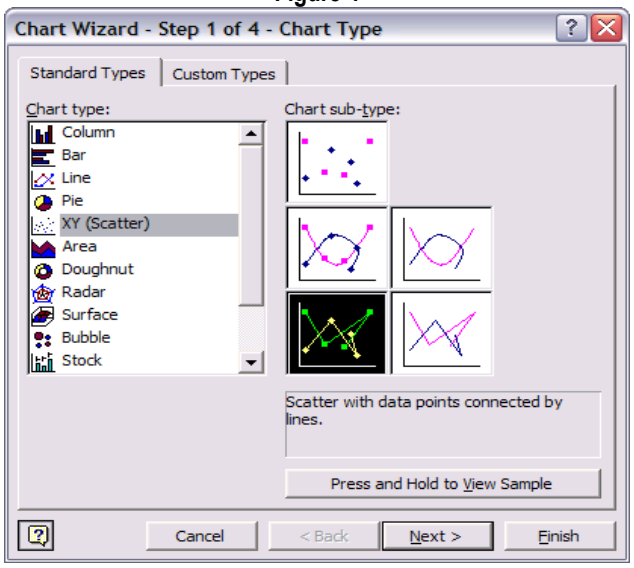
In this section, we will demonstrate through a case study how to apply the regression based techniques outlined above. For the purposes of this paper, we are assuming the Microsoft ® Excel Analysis Toolpak add-in has been installed. Please see Appendix A for instructions on installing the Analysis Toolpak using Microsoft Excel 2003.

Scatterplots – These graphs can be created without using the Microsoft ® Excel Analysis Toolpak, but require raw data. In the example below, for Iteration 3, the following segment of raw data is displayed.

Date	7/24	7/25	7/28	7/29	7/30	7/31	7/31	8/4	8/5	8/6	8/7	8/8	8/11	8/12	8/13
TOTAL	74	77	89	83	79	94	114	108	110	132	148	161	168	168	171

From this data set, we can create a Scatterplot of this data by using the "Chart Wizard" within Excel (See screenshot below for reference). Your independent variable should be along the X-axis and your dependent variable should be along the Y-axis. Again, the purpose of the Scatterplot is to assess a selected independent variable's predictive power by using it to predict the dependent variable by itself. Select the "XY (Scatter)" option to create your graph and select "Next". Follow additional instructions to create your graph.

Figure 4



Once your graph has been created, add a linear trendline to the graph by right clicking on the plotted line and selecting “Add Trendline”. Display the equation and R^2 on the chart by double clicking on the trend line, selecting the options tab and checking the boxes to “Display equation on chart” and “Display R-squared value on chart”. (See screen shots below)

Figure 5

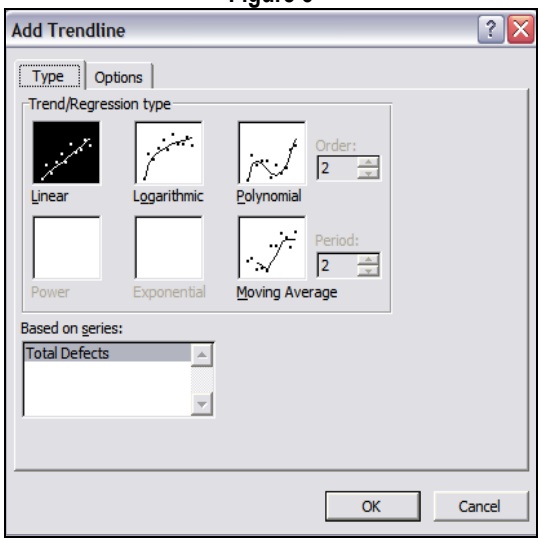
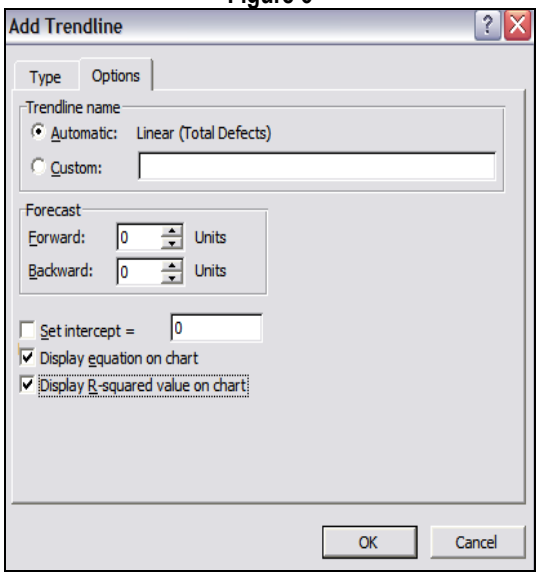
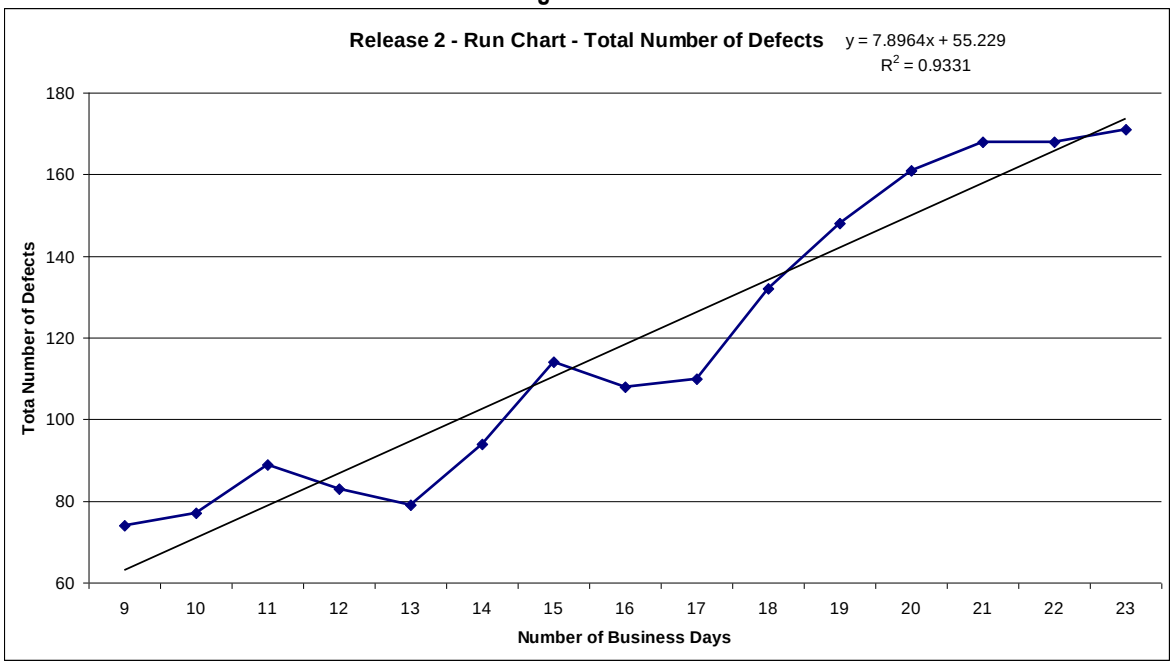


Figure 6



As the graph below illustrates, this will help to get an idea of the independent variable’s predictive power. A R^2 approaching 1 is a very good sign.

Figure 7 - Release 2:



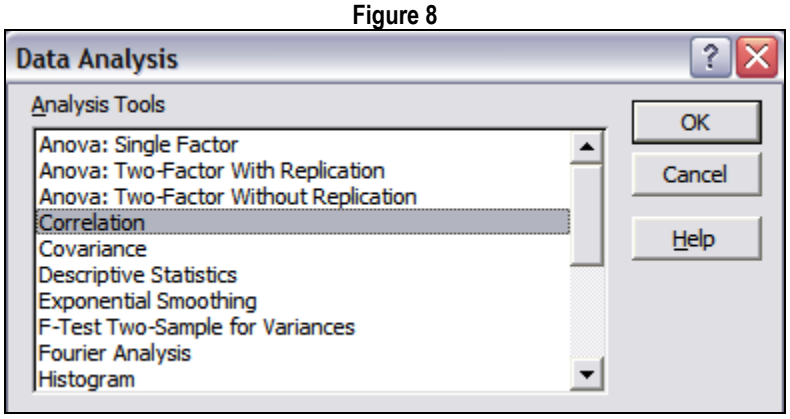
In the case of Correlation Matrices, the Microsoft ® Excel Analysis Toolpak add-in is required. In our case example taken from the insurance industry, the following raw data points gathered over five releases have been captured for the identified independent variables:

Raw Data

	Independent Variables			Dependent Variable
Release	Total Number of States in Release	Total Number of Days in Test	Complexity Scores	Total Number of Defects
1	4	24	10	353
2	2	23	7	171
3	6	39	6	129
4	2	33	6	68
5	2	20	8	377

Under “Data Analysis” in the “Tools” dropdown, select the “Correlation” function (See screen shot for reference). Select all your variables, both independent and dependent in the “Input Range”.

Figure 8



As stated in the "Analysis" section, the correlation matrix will give you an idea of how well the independent variables correlate to the dependent variable and each other. Using this matrix, you can begin to construct and tweak your model as necessary to make it as predictive as possible. The "Data Collection" section provides a roadmap for constructing a predictive model.

Using the independent and dependent variables outlined in the beginning of this section, the following correlation matrix was created:

	Total # of Business Days in Test	Total # of States in Release	Complexity Scores	Total # of Defects
Total # of Business Days in Test	1.00			
Total # of States in Release	0.66	1.00		
Complexity Scores	-0.67	-0.03	1.00	
Total # of Defects	-0.76	-0.10	0.87	1.00

Looking at the independent variables with correlations closest to 1 or -1 in relation to the dependent variable (seen in yellow), the following independent variables have been identified to predict the dependent variable (total number of defects in each release):

1. Total # of business days in test (-0.76)
2. Complexity scores (.87)

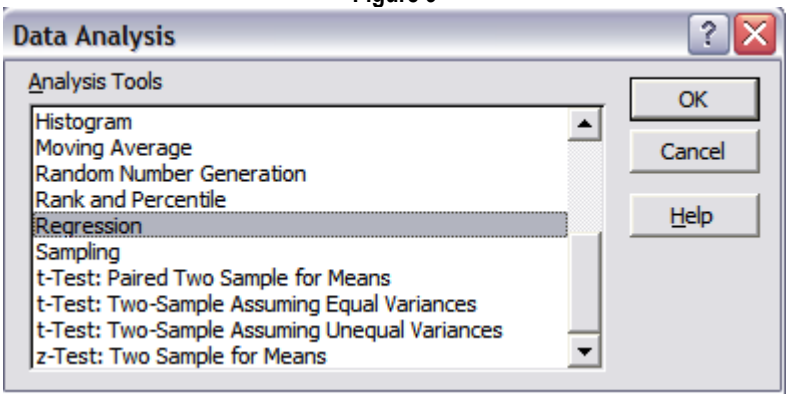
The strong negative correlation between "Total # of business days in test" and "Total # of Defects" does not seem intuitive. It's more reasonable to assume that as the number of days allocated to testing increases, the number of defects also increases, which would yield a strong positive correlation instead. It is in these

situations that the project manager must use judgment to determine what should be included in the regression model.

Given the limited amount of data available, the strong negative correlation may be due to random variation and therefore could ultimately reduce the predictive power of the regression model. Until more data elements can be gathered to confirm or refute this seemly counter-intuitive relationship, the "Total # of business days in test" will be removed from the model. The lesson here is to use your best judgment to not only include variables that have strong relationships with the dependent variable, but also ones that make logical sense. This leaves us with a single independent variable, "Complexity Score", from which to build our model.

To generate the regression model, under "Data Analysis" in the "Tools" dropdown, select the "Regression" function (See screen shot for reference). The independent variables are the "Input X-Range" and the dependent variable is the "Input Y-Range".

Figure 9



The regression model generated uses the one independent variable, "Complexity Scores", to predict "Total Number of Defects". Below is a summary of the results from the model. The full regression output can be found in Appendix B:

Summary of Regression Model

	Coefficients	P-value
Intercept	-308.84	.18

Complexity Scores (X_1)	71.41	.06
Regression Statistics		
Adjusted R^2	.67	
Standard Error	79.60	
F-Statistic	9.01	

Using our linear equation mentioned above in the "Regression" section:

$$\text{(Total Number of Defects) } Y = m_1X_1 + m_2X_2 + m_3X_3 + m_4X_4\ldots\ldots + b$$

The coefficients in the table above are used to generate the following equation:

$$\text{(Total Number of Defects) } Y = (71.4*X_1) + (-308.8)$$

By plugging in values for the independent variables, the model will calculate the predicted number of defects. The question then becomes for the project manager, "How reliable is this prediction? Can I use this model to confidently predict the total number of defects I'll encounter on my next release?"

The answer lies in the Adjusted R^2, the Standard Error, F-Statistic and P-values for the model.

The Good News

1. Independent variable p-value – The p-value for the independent variable is approximately .06, a good litmus test for statistical significance assuming a 90% confidence interval. Generally speaking, 1-p-value should be greater the identified confidence interval.
2. F-Statistic – Another measure of statistical significance, this metric is just above nine (9.01) with a significance of .058 (See Appendix B), indicating statistical significance assuming as 90% confidence interval. As mentioned in the "Regression" section, we want the F-Statistic to be as large as possible.
3. Adjusted R^2 - A measure of what our model has explained versus what it has not is at .66, meaning changes in our independent variable can explain 66% of the changes in our dependent variable.

Even Better If

1. Y-Intercept p-value – Yielding a p-value of .18, this p-value causes some concern, however this value isn't too far off our confidence interval of 90%. Gathering more data or adding another predictive independent variable will most likely improve this p-value.
2. Standard Error – The model yielded a standard error of 79, meaning the actual number defects, assuming a normal distribution, will be between +/- approximately 160 defects from the mean. As with the Y-intercept p-value, more data points or adding another predictive independent variable will help bring this number down.

Based on these results and interpretations, it can be concluded that this model has yielded an equation that while not perfect and has acknowledged limitations, can be used as a valuable input for estimation. In addition, it is important to point out that regression models will not exist in a vacuum; other project management tools can be used in addition to the D.A.R.E framework and as more data points are collected, the project manager can become even more confident with the estimates he or she are providing. Project mangers will know that they have removed as much guesswork as possible from their models because they have D.A.R.E'd to be different.

Conclusions

Mathematic assets, methods and capabilities help to develop predictive analytics and business optimization to create solutions for the challenges we face everyday as project managers. Discover predictive insights and turn that into operational reality to close the gap between strategy and execution. Going forward, here are some helpful takeaways from the D.A.R.E framework.

1. Harness the value of D.A.R.E. It provides a proven statistical modeling technique and an objective approach to use to generate estimates.
2. Understand the limitations of D.A.R.E; data sets will be typically small. Several releases are required to achieve a statistically significant model. While not explored in this paper, other research opportunities exist in extending the D.A.R.E. framework to incorporate data from multiple

similar projects in an effort to reduce the number of iterations needed to achieve a more accurate model.

3. Gather the information. Software projects have a lot of great data. Take advantage of it and use it to benefit your success. Predictions are not perfect but they allow you to set goals and provide meaningful metrics.
4. Think! Using intelligent information is better than guessing, using intuition, or simply looking at the averages.

About the Authors

Luke Kelleher is a Project Executive and Senior Managing Consultant with IBM's Global Business Services Insurance practice. His specialties include complex systems integration, project management, data conversion and object-based project methodologies. He can be reached at luke.kelleher@us.ibm.com.

David Lipien, PMP is a Senior Managing Consultant with IBM's Global Business Services Insurance practice. His specialties include complex systems integration, release management, internet-based technologies, wireless technologies and object-based project methodologies.
He can be reached at lipien@us.ibm.com.

Seth Newell is a Senior Consultant with IBM's Global Business Services Insurance practice. His specialties include software package implementations and process improvement. He can be reached at srnewell@us.ibm.com.

Justin Ryan is a Managing Consultant with IBM's Global Business Services CRM practice. His specialties include software package implementations, project management, and custom application software development. He can be reached at justryan@us.ibm.com.

Contributors

This paper would not have been possible without the insights and contributions of Sebastian Purakan, Gordana Radmilovic, Shradha Gokhale, Nick Concha, Tom Ruggieri and the great team of software developers and management consultants we have all worked with over the years.

Get Products and Technologies

To learn more about IBM Rational products, visit the developerWorks Rational zone. You'll find technical documentation, how-to articles, education, downloads, product information, and more.

ClearQuest users and administrators can find more resources in the ClearQuest section of the developerWorks Rational zone, including ClearQuest hooks, Eclipse plug-ins, product documentation, articles and whitepapers.

References

1. Microsoft. (2008). Microsoft.com. Retrieved December 26, 2008, from http://office.microsoft.com/en-us/excel/HP011277241033.aspx
2. Microsoft. (2008). Microsoft.com. Retrieved December 26, 2008, from http://office.microsoft.com/en-us/excel/HP100908421033.aspx
3. Project Management Institute Body of Knowledge (PMBOK) at pmi.org.
4. Carnegie Mellon Software Engineering Institute. (2008). Retrieved April 26, 2009, from http://www.sei.cmu.edu/cmm-p/version2/part1.pdf
5. Ideas from July 2007. (2009) IBM.com. Retrieved April 5, 2009, http://www.ibm.com/ibm/ideasfromibm/us/math/070907/images/IFI_070907.pdf
6. Anderson, D., Sweeney, D., Williams, T. Essentials of Modern Business Statistics with Microsoft Excel, 2nd Edition. New York: South-Western Educational Publishing.

Appendix A

As mentioned above, the Analysis Toolpak is an Excel add-in (add-in: A supplemental program that adds custom commands or custom features to Microsoft Office) program that is available when you install Microsoft Office or Excel. To use it in Excel, however, you need to load it first.

1. On the Tools menu, click Add-Ins.
2. In the Add-Ins available box, select the check box next to Analysis Toolpak, and then click OK. Tip: If Analysis Toolpak is not listed, click Browse to locate it.
3. If you see a message that tells you the Analysis Toolpak is not currently installed on your computer, click Yes to install it.
4. Click Tools on the menu bar. When you load the Analysis Toolpak, the Data Analysis command is added to the Tools menu.

Appendix B

Regression Statistics	
Multiple R	0.87
R Square	0.75
Adjusted R Square	0.67
Standard Error	79.60
Observations	5

	df	***SS***	***MS***	***F***	***Significance F***
Regression	1	57114	57114	9.01	0.058
Residual	3	19009	6336	0	0
Total	4	76123	0	0	0

	Coefficients	***Standard Error***	***t Stat***	***P-value***	***Lower 95%***	***Upper 95%***	***Lower 90.0%***	***Upper 90.0%***
Intercept	-308.84	179.58	-1.72	0.18	-880.33	262.65	-731.44	113.77
Aggregate Complexity in Each Release	71.41	23.79	3.00	0.06	-4.28	147.11	15.44	127.39