

The 3 Year Obsession: Length of experience the sole determinant of competence?

Meghana Bhagwat, Cytel, Pune, India
Manjusha Gode, Cytel, Pune, India

ABSTRACT

A technically strong team defines the identity of any functional service provider in the clinical programming arena. However measuring this strength is a rather tall order and hence elusive. The number of years of experience has become the sole indicator of quality even if as a fallback. And in that, companies tend to sometimes settle on numbers like 3 or 5. In emerging markets like India, where clinical research is still in its nascent state, this poses a challenge since clinical trial programming is not yet a widely known field and hence availability of experienced resource is limited. The challenge is to provide the same quality provided by experienced programmers with available resources. This paper will discuss a few methods that have been effectively used to bridge the gap between expectations and availability to provide high quality deliverables with the available programming pool.

1. INTRODUCTION

Over the past decade or so, drug and device development sponsors have started considering functional service providers (FSP) for outsourcing ventures. Their major concern however is ensuring quality deliverables. This also holds true when global companies are looking to transfer workload to their own offshore group as a cost effective measure.

Technically strong and experienced teams are the key to providing superior deliverables. This holds especially true for statistical programming groups since a well trained and skilled programming unit is their core asset.

Measuring quality of programming resource however, is a difficult task. Over a period of time, the number of years of experience has become one of the indicators of quality, even if as a fallback mechanism. As a result, various organizations tend to settle on numbers like 3 or 5 years.

This poses certain challenges in countries like India for the following reasons:

- Clinical research is still in its nascent stages.
- Clinical trial programming is not a popular career option as yet.
- There is a shortage of good educational institutes which specifically impart knowledge of clinical trial programming.

Hence availability of trained and experienced resources is limited.

Even with the few available resources with required number of years of experience, there are some additional challenges that functional providers need to work with. FSP teams do not get exposure as compared to sponsor teams as described in table 1.

PhUSE 2012

Table 1

	Sponsor teams	FSP Teams	Reasons	Outcome
Type of programming handled	Efficacy and complex programming.	Safety and simple programming.	• Easy to co-ordinate simpler tasks • Volume of safety programming is higher.	Experience can remain limited to safety and TLF programming.
Cross-functional exposure	Interact with clinical scientists, data management and medical writing.	Interact with members from statistical programming team only.	• Time zone differences. • Geographic location differences.	No interaction with stakeholders leaves FSP programmers with a narrow view of the big picture.
Decision making	Work in conjunction with other teams and statisticians. Involved in decision making.	Get informed about decisions	Same as above	FSP team does not get a forum to discuss study details or rules. Hence they do not develop an approach of thinking objectively about specifications and adding individual input.

As a result of the factors mentioned above, FSP teams have limited exposure to trial programming in terms of both time as well as quality. The global client or on-shore counterparts comprise of programmers with more than 7-10 years of experience and a vast knowledge base. Hence the disparity between sponsor expectations and FSP team's expertise becomes more pronounced. The challenge then, is to meet the expectations set with the available resource pool.

Essentially the task is to:

1. Improve performance of programmers with **less than 3 years** of experience to match up to their more experienced counterparts
2. Build expertise of programmers with **more than 3 years** of experience but with limited exposure

To achieve this goal it is essential to provide training in clinical programming with an accelerated learning curve. Along with the technical know-how, to build well-rounded professionals who add value, it is necessary to impart a big picture view of the clinical research industry as well.

To simplify the training process we have divided the required skill set of statistical programmers as described in the following table.

Table 2

	Required Expertise
Programming skills (SAS, R or other software applications)	• Application of SAS techniques • Programming concepts for standard definitions
Clinical Domain knowledge	• In-depth knowledge of the clinical trial process • Knowledge of the data collection process • Understand the protocol and SAP as well as underlying assumptions. • Therapeutic area concepts.
Ability to handle complex programming	• Exposure to different types of reports and the rationale behind them • Understand the impact of error for the output produced by programs • Macro development • Efficacy analysis
Communication skills	• Communicating questions and issues effectively • Documentation needed for studies like dataset derivations
Sponsors anticipate that programmers with at least three years experience will have a minimum rating of 3 (Good) in above skills. Our goal is to bring programmers with less than 3 years experience to the required expectation.	

PhUSE 2012

We have found a few methods to be successful in bridging this gap between expectations and availability to develop high quality resources.

The training process described here is not meant to be a short cut to experience. Instead it is a directed and dedicated effort to augment the skill set of bright young programmers in order to help them succeed at their job and at the same time overcome the lack of availability of experienced resources.

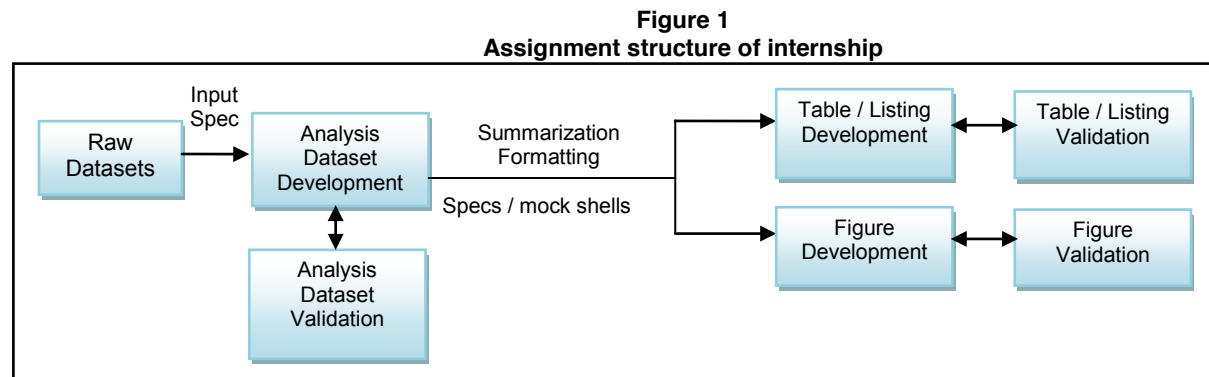
2. THE TRAINING PROCESS

2.1 INTERNSHIP FOR ENTRY-LEVEL PROGRAMMERS

To address the lack of educational institutes which provide focused attention on clinical trial programming, a unique internship program was developed around 3 years ago. It is run by experienced programmers within Cytel and focuses on statistical programming with reference to the clinical domain. Lasting 4 months, this internship is designed for entry-level programmers who are trained in or are familiar with use of BASE SAS.

The overall structure of the internship is described below:

- The complete clinical trial programming process is covered as shown in figure 1. Each task is developed as an assignment lasting about 2 weeks.



- Participants are provided with programming guidelines as good practices as well as additional background material.
- They are also provided with a list of tasks to be performed which programmers would normally do on a live study like reading study documents, maintaining issue logs etc.
- Regular meetings in the form of teleconferences are held with the trainer to get questions answered. This gives the students a feel for real world situations where the only way to communicate with global teams is via telemedia.
- After submission of their programs, their code is checked for accuracy of specifications, efficient coding practices as well as understanding of specifications.
- Detailed individual input is provided along with time for participants to re-submit code after implementing suggested changes.
- Grades are provided at the end of each assignment.
- The assignments are developed using test data for common data domains like laboratory, vital signs, adverse event etc such that the participants are exposed to a variety of clinical trial concepts. The data is simulated to replicate real world scenarios.

Successful participants at the end of the internship are interviewed and assimilated into the company's programming pool if found suitable.

The internship is meant specifically for entry level programmers before hiring them. Irrespective of prior experience, once new programmers join the Cytel team, additional training is provided before assigning them to live projects. This training is based on each individual's assessment and consists of the following methods.

2.2 MOCK ASSIGNMENTS

Mock assignments are programming assignments developed by Cytel programmers for various experience levels. Dummy data is provided along with a specification document and part of the study details.

Some examples of the assignments are:

- SDTM specifications development and migration
- deriving lab CTC grades
- developing generic macros

PhUSE 2012

- macro validation
- standard safety TLFs
- analysis datasets

When a new member joins the programming team, their supervisor assigns them multiple assignments with timelines associated. Regular meetings are held to discuss questions that the programmer may have. After completion the supervisor or designee performs a complete review of the submission and provides in-depth feedback to the programmer.

The purpose of this exercise is two-fold:

1. It gives the supervisor an insight into the exposure the programmer has received to date. This helps in deciding which project would be the best fit for the programmer to begin with.
2. Secondly there are cases where the supervisor needs the programmer for a specific project which requires skills that he/she has not been trained on. This works as hands-on training for the programmer and builds their comfort level before being assigned to production work.

A common example of this is, lately a large number of sponsor companies are shifting towards migrating data to CDISC standards. But availability of programmers with CDISC knowledge is limited. So an assignment to address this includes, annotating a few CRF pages using CDISC SDTM guidelines and migrate corresponding legacy data to SDTM standards using the annotations. This assignment helps programmers to not only get trained on SDTM guidelines but also gives them exposure to implementing the guidelines as well. At the same time it gives their supervisor insight on how quickly and effectively the programmer can pick up a new skill

These assignments are highly customized to each individual's requirement and expectations from them. Based on outcome of one assignment the next one is assigned. The time that programmers spend working on mock assignments can vary from two weeks to two months.

2.3 TRAINING

To further add to the skill set, periodic training programs are also conducted for all programmers. These cover a variety of topics from the basics to more advanced statistical concepts as well as therapeutic area based sessions.

2.3.1 INTERNAL TRAINING

Some of the training programs which have been developed in-house by Cytel programmers and statisticians include:

- CDISC SDTM: a 5 day course with a 2 hour session on each day which walks the participants through the implementation guide as well as the model. Each day includes various hand-on exercises which everyone is expected to complete, either alone or in groups
- AE: this 4 day course with a 2 hour session on each day talks about adverse events in detail starting from the collection stage to reporting. This interactive training discusses everything from definition of adverse events and their properties as well as programming concepts like date imputations. Exercises for writing code for these definitions are also included as well as for creating some of the standard AE tables.
- Programming and statistical concepts in Oncology: this week long training has been developed by statisticians. The theory behind survival analysis is discussed. Along with that programming techniques for oncology specific derivations like censoring flags and time to events are explained. It also introduces them to the common study designs, efficacy end-points as well as data structure in oncology trials.

Meetings are conducted to interactively discuss best solutions for every exercise. A couple of weeks after the training an hour long review session is conducted which summarizes the training along with some team exercises. Feedback is provided as well as any project specific questions are answered.

2.3.2 EXTERNAL TRAINING

Along with internal training, external training by industry experts is also conducted to impart programming and therapeutic area concepts. Soft skills training programs are included as well.

Some standard training programs include:

- ADaM guideline implementation
- Define.xml
- Oncology concepts
- Communication and presentation skills

2.3.3 CONFERENCE PARTICIPATION

Programmers are encouraged to participate in national and international industry specific conferences to present new ideas. These conferences provide programmers with a wide forum to interact with peers from different organizations across geographic regions thus imparting a global mindset.

2.4 PRODUCTIVITY TOOLS

Programming teams are enabled by an effective set of productivity improvement tools. To that effect Cytel has an automation team which develops tools to automate some of the more repetitive tasks. This team works like any other tool development team with standard processes to decide specifications, validation and user testing of tools, releasing versions and bug fixes.

Many of the programmers who contribute to the automation team have less than five years of experience. Also working in this team is open for anyone who has interest and down time in their trial programming projects.

The dual advantage of this team is:

- Productivity tools with company-wide usage get built without extra cost of having a full-time automation team.
- Programmers get experience in generic programming techniques thus imparting them with an additional skill set.

Even though every programmer doesn't work as a part of the automation team, the reason for including it here as a training tool is that the team gives programmers exposure to working on macro development which imparts specific skills like defensive coding and writing efficient code.

2.5 GROUP DISCUSSIONS

Regular group discussions are conducted to discuss clinical trial concepts, protocol designs, therapeutic area knowledge and programming techniques. These groups consist of 4-5 programmers along with a mentor.

These discussion groups are voluntary and organized by programmers. They are provided with a basic curriculum but are encouraged to add topics based on their project requirements or interests. Discussions are strictly theoretical and are aimed at providing additional details into the finer nuances of the clinical trial arena which is usually gained through years of experience.

Some of the topics discussed in these sessions include:

- Adverse event date imputation rules and the rationale behind them.
- Missing value imputation rules especially for efficacy data as well as imputation techniques like last observation carry forward and their rationale.
- Baseline and change from baseline definitions
- Lab visit windows and their assignment. Deciding analysis values when multiple visits fall in the same window.
- Dosing schedules for parallel and cross-over studies
- Common study designs used in various therapeutic areas.
- Validation tips and techniques.

These groups are led by a mentor, usually a programmer with many years of experience in statistical programming. They work as moderators as well as experts to answer questions. The programmers are the ones who participate in the discussion and bring various points to the table based on their projects as well as through reading various books, protocols and other research material. Questions are raised and the group aims to find answers. Many of these questions are raised to the on-shore team as well. Sometimes study specific needs are also covered at conceptual level during these sessions. These kind of knowledge sharing discussions help to have quick and correct resolutions for programmers for their study related questions.

2.6 PROJECT MANAGERS AS TECHNICAL LEADS

Leads with more than 7 years of experience in statistical programming in clinical trials work as project leads. These leads either have statistical training and many have exposure of working outside India.

Their role is designed such that they are not directly handling programming tasks. Instead their role involves:

- Project management and resource allocation. They ensure timely and accurate submission of deliverables either to the client or internal global teams.
- Full time technical support including programming help and answering questions based on SAP and protocol as far as possible.
- Full output review which is conducted by visual checks, usage of macros, cross-table checks etc.
- Code review for complex or programs with outputs ranked as high risk.

PhUSE 2012

- Work as point of contact with client as well as counterpart teams located globally
- Act as mentors for group discussions described earlier
- Develop and conduct internal trainings described earlier
- Provide soft skills trainings like communication skills.

Project managers act as a buffer between the team and the customer ensuring quality and timeliness of deliverables. This relieves pressure off of juniors.

Using the reference of Table 2 above, Table 3 summarizes the gaps in expectation and our way of providing a solution to build required expertise as discussed so far.

Table 3

Required skill	Way to address gap	Step
Improve technical skills	Exposure to programming techniques beyond available projects	Mock assignments, full time support by experienced lead
Improve domain knowledge	Provide theoretical knowledge	Group discussions, external training, support by project leads
Improve complex programming capacity	Exposure to programming like efficacy analysis, statistical analysis and macro development	Mock assignments, macro and tool development projects
Improve communication skills	Provide opportunities to address identified problem areas	Soft skills training, attending and presenting at conferences, participate in discussions with other programmers.

3. EFFECTIVENESS OF THE TRAINING PROCESS

To demonstrate the effectiveness of these methods, we collected some data internally from line managers and project leads. They were asked to provide ratings for their team members who have undergone training, based on the specific skills expected from programmers with at least 3-4 years of experience. Every effort has been made to provide ratings using a consistent criteria and irrespective of the programmer's actual experience. Ratings in each of the expected skills are compared against experience level as shown below.

Table 4

		Level 1 N=43	Level 2 N=30
Technical	Mean	3.3	3.8
	Median	3.0	4.0
	Min, Max	2, 5	2, 5
	Q1, Q3	2.0, 5.0	2.0, 5.0
Domain	Mean	3.1	3.9
	Median	3.0	4.0
	Min, Max	2, 5	3, 5
	Q1, Q3	1.5, 4.5	2.5, 4.5
Complexity	Mean	3.1	3.6
	Median	3.0	3.5
	Min, Max	2, 5	1, 5
	Q1, Q3	1.5, 5.0	1.0, 4.5
Communication	Mean	3.1	3.8
	Median	3.0	4.0
	Min, Max	2, 5	3, 5
	Q1, Q3	2.0, 4.5	2.5, 4.5
Ratings: 1 = Unsatisfactory, 2 = Satisfactory, 3 = Good, 4 = Very Good, 5 = Excellent N = number of programmers with available ratings Level 1 : <= 3 years, Level 2: > 3 years			

Figure 2

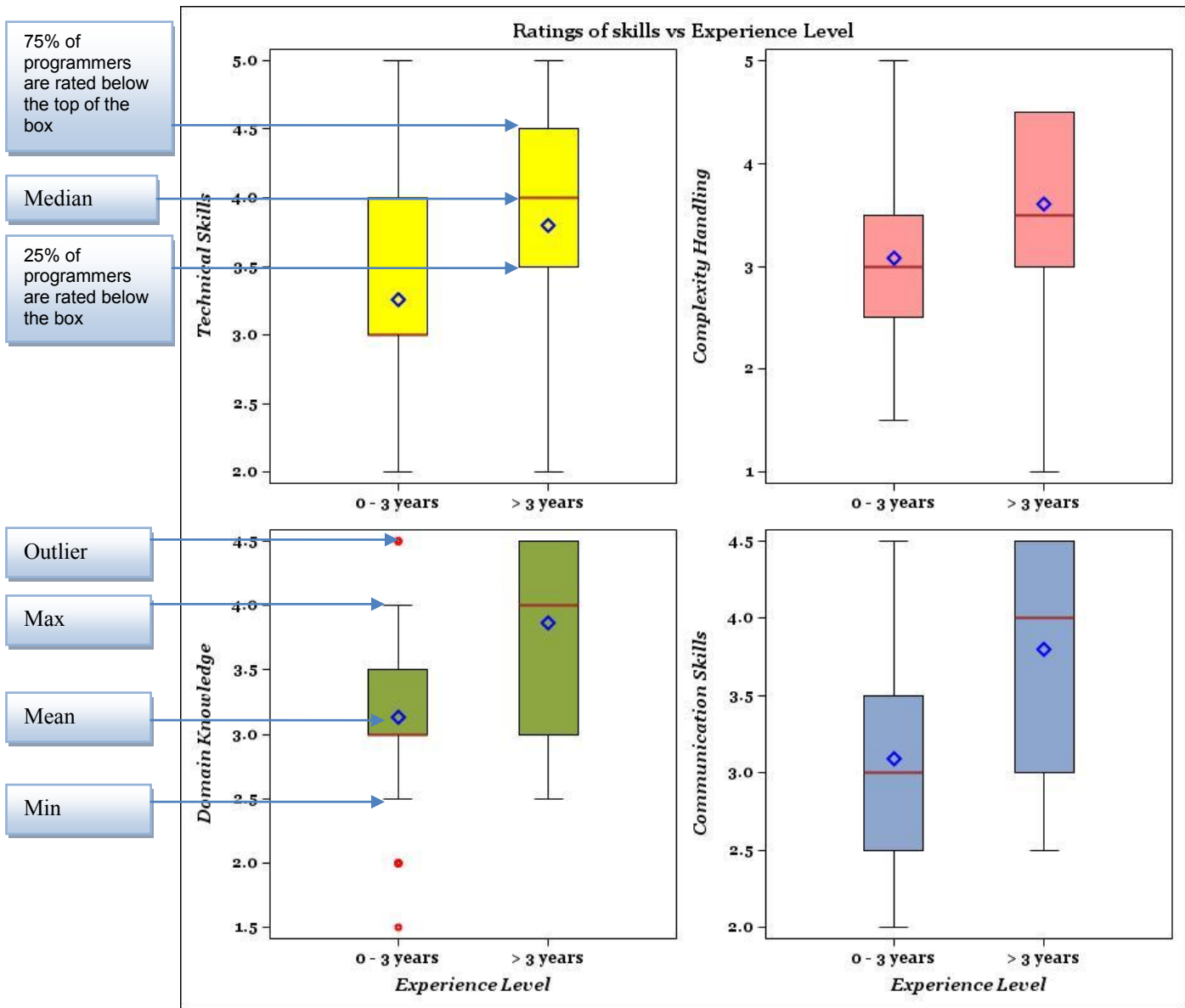


Table 4 and figure 2 both demonstrate that:

1. The average ratings of programmers with less than 3 years of experience are above 3, which is classified as good.
2. Average ratings of programmers with more than 3 years of experience are higher than the juniors, which is expected.
3. Level 1 programmers are rated at a mean of 3 ('Good') is evidence of the premise that with a directed training mechanism, it is possible to have a relatively young team perform at more than satisfactory levels.
4. Outliers indicate that aptitude also plays a role in a programmer being successful.

Figure 3

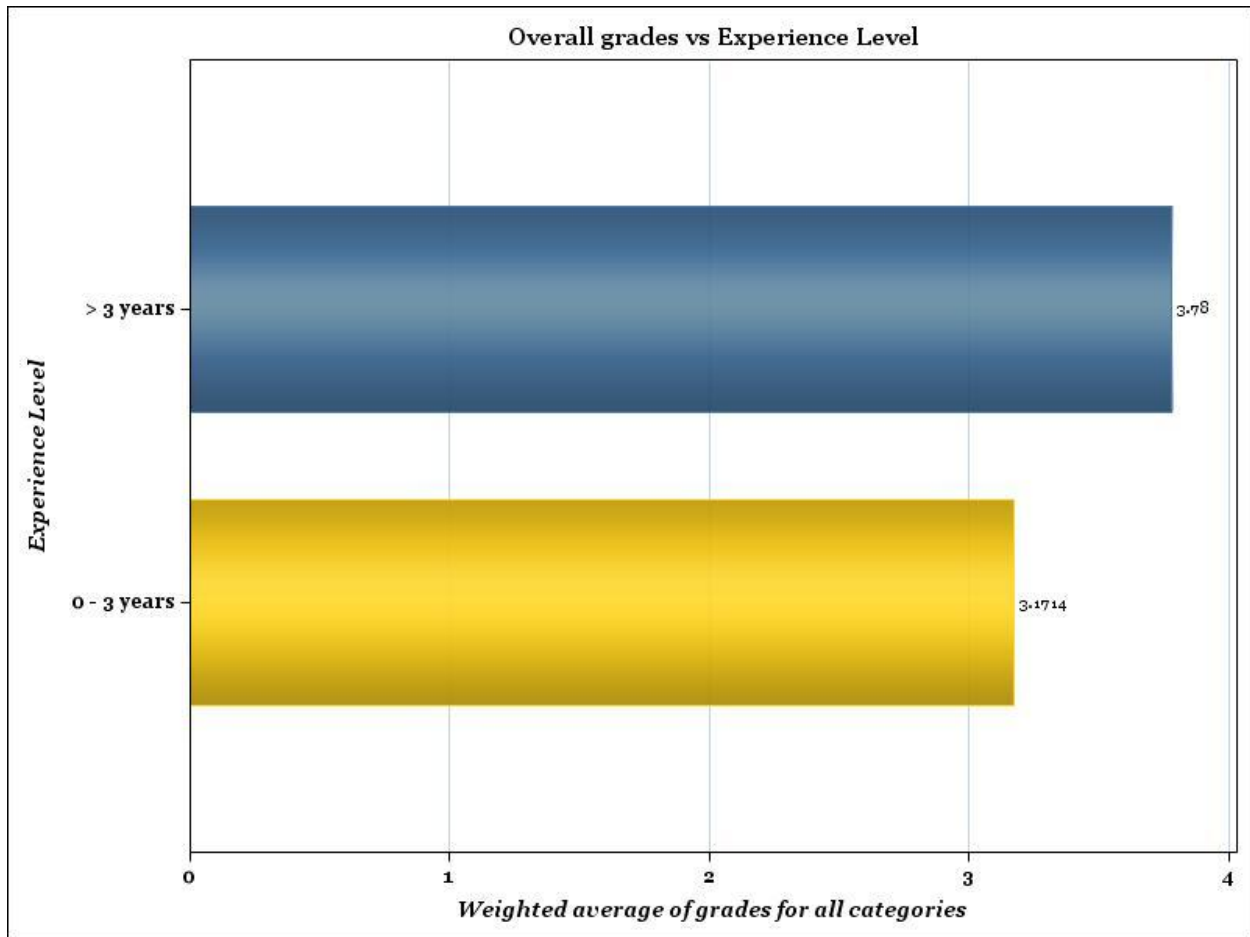


Figure 3 shows the mean of overall rating of programmers across all experience levels. The mean is a weighted average of all four skills described by figure 2. Here, it is seen that level 1 programmers are at a weighted average of 3.1 which is not too separated from the 3.78 for level 3 programmers.

This data is not sufficient to conclusively make a statement that the training process will yield 100% results for every single programmer. However the fact that the average rating is above satisfactory, is encouraging for us. To improve results, we continue to fine tune various aspects of the process.

4. SUCCESS STORIES

To further elaborate the effectiveness and highlight the success of the training process, a couple of stories are described below. These explain how we were able to change customer perspective regarding team requirements. All programmers mentioned here have undergone the training process.

4.1 MACRO DEVELOPMENT FOR ONCOLOGY EFFICACY DATA:

Client: top 10 pharmaceutical company with team located in the US

Team Requested: 2 senior macro developers with more than 10 years of experience in clinical domain.

Available FSP Team: 1 project manager with 7 years experience and 2 programmers with 1.5 years of experience each and no oncology background.

Task: Develop a standard macro to derive efficacy dataset with RECIST criteria definitions using client standard data structures.

Plan of action:

- Intensive training and review of existing data structures, oncology basics and macro development methods
- Programming with full time support by project manager. PM would sit alongside programmers to outline code as well as provide SAS tips and techniques
- Simultaneous in-depth review of code and macro validation

PhUSE 2012

Outcome: Successfully delivered on time with desired quality alongside accolades from client team. More importantly both junior programmers were specifically requested for a subsequent macro development project despite less experience.

4.2 SDTM MIGRATION AND ISS/ISE:

Client: mid-size pharmaceutical company

Team requested: 2 senior programmers with at least 7 years and 2 mid-level programmers with 4-5 years of experience.

Available FSP team: 1 lead programmer with 2.5 years of experience and 4 programmers with less than 2 years experience

Task: Migration of raw data to SDTM standard datasets for six, phase II studies, pooling of data to develop ADAM datasets as per CDISC standards and develop TLFs as per ISS SAP.

Challenge: Unavailability of any project manager/experienced team lead to provide support due to prior project allocations.

Outcome: Again, the team was able to provide timely and accurate deliverables with accolades from the client team and statistician. The lead programmer with 2.5 years of experience was requested by client team for their next project.

5. CHALLENGES:

This entire process is not without some challenges. Some of the more prominent ones are:

- External training expenses.
- Requirement of significant time to be spent by experienced resources in development and delivery of training
- Some of the more talented and experienced resource who work as project leads cannot be available for programming and hence are not billable to clients
- Significant amount of time being spent on review
- Team size tends to be larger than ideal.
- High pressure timelines and project needs are always deciding factors for any FSP to allocate resources. Hence some new programmers need to start on live projects right away. They participate in the training process on an ad-hoc basis thus reducing their exposure to comprehensive training.

6. CONCLUSION

All the above challenges do factor into the cost of execution. However there is a lack of required experience and skills in currently available resources. To align with company needs therefore, having a comprehensive training process is one of the best options for functional service providers to successfully execute projects. As demonstrated having a well rounded training approach can improve the overall effective experience of the statistical programming group. This aids in improving the quality of output delivered and consequently results in happy clients.

ACKNOWLEDGEMENTS

We thank our colleagues at Cytel who took the time to review the paper and provide us their valuable feedback and data.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Meghana Bhagwat
Cytel Statistical Software and Services Pvt. Ltd.
Pune, India
meghana.bhagwat@cytel.com
www.cytel.com

Manjusha Gode
Cytel Statistical Software and Services Pvt. Ltd.
Pune, India
manjusha.gode@cytel.com
www.cytel.com

Brand and product names are trademarks of their respective companies.