Lean Forecasting

Forecasting Through Uncertainty and Speaking Value Rather Than Cost

LKNA 2013 Tutorial by Troy Magennis
What you will learn….  
(in order of survey responses)

• How to forecast likely delivery dates and costs
• How to calculate Cost of Delay and why it matters
• How to quantify the benefits of decreasing cycle time in cash-flow and throughput
• How to calculate the cost and date impact of defects and blocking events
• How to analyze systems for staff imbalance
• How to analyze and use historical data when available
• What constitutes "enough" historical data and what to do when you have less (or none)
What topics or questions did we miss. List three more topic areas

• Story points and velocity (Scrum) relationships
• What the pre-requisites to getting a believable delivery schedule
• (Automated) Tools that help with calculations and predictions
• Card size -- do they all have to be the same?
• Little's law and WIP and its relationship with prediction
• I'm creating a new team for a new project, am I screwed when my boss asks for a schedule?
Any general requests or questions you have

- Please give us a lot of notes to take away. Tools that we can use. And a contact email that we can validate our learnings without fear.
- Monte Carlo analysis is great, but can you present other statistical/quantitative techniques to base predictions on?
- I'm really interested in the technique, but equally important is the hard part: Having the conversations with the right folks in the right way to get them to use the math responsibly and accurately. I'd love help with both!
Agenda – Day 1

- 9:15 Start of festivities
  - Agenda, goals and introductions
  - Introduction Forecasting and Modeling
- 10:00 5 minute break
  - Current estimation and forecasting
  - Installing and using KanbanSim
- 11:00 5 minute break
  - Exercises: basic forecasting
- 11:45 Lunch

- 12:45pm Lunch
  - Cost of Delay
  - Cycle Time Shape
  - Mining Cycle Time Data
- 2:00 5 minute break
  - Forecasting using Historical Data
  - Exercises
- 3:00 15 minute break
  - Staff role forecasting
  - Exercises
- 4:00 5 minute break
  - Wrap-up and recap
- 4:30 End of festivities
Introduction to Forecasting & Modeling
Forecasts are answers to questions about event(s) that haven't occurred yet.

Cost of Delay: $1,234,000

15th August 2013

Staff
If a measurement changes over time or is different each time you measure it, it is a DISTRIBUTION
Exercise – 15 Minutes

• Describe some of the questions you commonly get asked to answer
• Describe how you currently answer these questions
• Describe the process used throughout your organization in portfolio planning and project prioritization
• Survey Results....
Describe what forecasting and estimating process you use now

- Little to nothing
- Calculate throughput and lead time. Calculate variance of the same. Use these to forecast early, likely and late dates for upcoming stories.
- On Kanban projects: Service level-based estimating as work is added to backlog, weekly/on-demand prioritization of what few items are next-up. We do monthly retros to review reports on reliability, WIP and throughput, by service level and work types, to inform forecasting. On Scrum projects, we take teams through a planning day to prioritize work items for next sprint (2-4 weeks), break them into stories, estimate by fibonacci series-based story points, then commit to which items will be delivered based on velocity history and forecasted team capacity. I would like to introduce Kanban approaches to improve this wasteful and rigid process that drives dysfunctional team behaviors around meeting story point commitments instead focusing more on business value.
- As a CMMI level 5 company our projects use various performance baselines and models (created from historical data)
- We use a process that combines intuition and cost estimates and ROI projections.
- We invest a lot of effort up front into trying to get a project into roughly equal cards. We use a cards/dev/month ballpark informed by historic data on other projects and then apply a 30-40% contingency factor for new scope, bugfixing, cards we've ID'd that split into cards because our first pass was too large. So 1 dev doing 1.65 cards/month with a project scope of 10 cards and a 30% contingency == 13 cards, 8 months of work.
- Attempt to remove as many estimates as possible.
Describe the types of forecasting / estimating you need to fulfill

- What improvements can I start predicting in the near future in terms of lead and cycle time when they don’t even have a Kanban board and have been doing Scrum?
- When will "x" be in production?
- Coming from a few years practicing Kanban, where we use service levels to broadly estimate work, I am moving to a scrum project next. I want to try to nudge our company’s scrum process to apply some Leaner estimating techniques that I read about in Corey Ladas' scrumban book: such as reducing estimating waste by standardizing story sizes, in order to be able to forecast throughput instead of story point velocity, and to move towards just-in-time sequencing of work, instead of batch planning with scope commitments at the beginning of each sprint. Wondering if you will have experience to share about the impacts on forecasting and estimating when moving scrum organizations towards the leaner Kanban model of continuous planning and just-in-time estimating, without the goal to convert scrum projects to kanban.
- Based on performance of a project until now, what is the likely delivery date, and the last revised estimate compared to budget
- We have visualized the work in our portfolio management system. And we have a rudimentary prioritization process in place. I want to add more discipline to that process with cost of delay and use of economic measures to emphasize flow and small batches. Being able to do what-if analysis would be a bonus.
- When will I get this particular feature, I need to know because of marketing lead time, advertising purchase, the rest of the company needs to know, etc. * When will this project, or this phase of the project be done? * If I get you more money for people how much quicker can I get this set of cards * I have a vague idea/project, how long will it take (not broken down into cards)
- How to move to probabilistic model of forecasting
A model is a tool used to mimic a real world process.

Model are a tool for low-cost experimentation.
Black Box Modeling

Full Step by Step Modeling
Guess

Model

(backtest)

(Model)

(update)

.measure

(reduce uncertainty)
All Models and Forecasts

• Mimic a complex process and therefore –
  – Are “wrong” until the final outcome is known
  – Are “best” when multiple methods are compared
  – Should be Back Tested

• Complex systems have “Chaotic” results –
  – Meaning small input changes can have a non-linear impact of the result
  – Butterfly effect...
    • Rounding errors after 5th decimal place in weather
Cycle Time Forecasting

1. Easy to Capture Metric
2. Forecasting using Historical Data (even a little)
3. Follows Known Distribution Pattern
4. Forecast at Project or Feature or Story Level
Can’t I just use Average Cycle Time?

<table>
<thead>
<tr>
<th>Forecast (Days)</th>
<th>Median</th>
<th>Mean</th>
<th>MC &gt;50%</th>
<th>MC &gt;85%</th>
<th>MC &gt;95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>525</td>
<td>670</td>
<td>677</td>
<td>793</td>
<td>851</td>
<td></td>
</tr>
</tbody>
</table>

Projection of cycle time to complete 25 stories in a sequential fashion based upon historical cycle time data.
1. Historical Cycle Time

2. Planned Resources/ Effort

3. The Work (Backlog)

<table>
<thead>
<tr>
<th>Backlog</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature 1</td>
</tr>
<tr>
<td>Feature 2</td>
</tr>
<tr>
<td>Feature 3</td>
</tr>
</tbody>
</table>

4. Historical Scope Creep Rate

5. Historical Defect Rate and Cycle Times

(optional)

Monte Carlo Process = Process to Combine Multiple Uncertain Measurements / Estimates
### Historical Story Lead Time Trend

![Histogram of Story Lead Times](image)

### Sum Random Numbers

<table>
<thead>
<tr>
<th>Random Numbers</th>
<th>Sum:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>295</td>
</tr>
<tr>
<td></td>
<td>410</td>
</tr>
<tr>
<td></td>
<td>187</td>
</tr>
</tbody>
</table>

### Basic Cycle Time Forecast Monte Carlo Process

1. Gather historical story lead-times
2. Build a set of random numbers based on pattern
3. Sum a random number for each remaining story to build a single outcome
4. Repeat many times to find the likelihood (odds) to build a pattern of likelihood outcomes

### Total Days

\[
\text{Total Days} = \frac{\text{Sum } (\text{Story}_n \times \text{Random}_n)}{\text{Effort}}
\]

### Days To Complete

![Histogram of Days to Complete](image)
Commercial in confidence

**Project Model**
- Staff, Work/Stories, Risks, Process, etc.

**Historical Data or Expert Judgment**

**Simulation Engine**

**Date Forecast**
- Date
- Certainty
- Risk Impact

**Budget Forecast**
- Cost
- Certainty
- Risk Impact

**Staff Analysis**
- Skill Balance
- Retention Risks
- Best Utilization

**Sensitivity Analysis**
- Factors to manage next
- Factors worth investigating

**Experiment / Update Loop – Change Model and Test Hypothesis**
How I Present Cycle Time Forecasts

When will my feature ship?

<table>
<thead>
<tr>
<th>Id</th>
<th>Current Team</th>
<th>Full Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>29 Jul</td>
<td></td>
<td>Feature 1</td>
</tr>
<tr>
<td>2</td>
<td>16 Aug</td>
<td></td>
<td>Feature 2</td>
</tr>
<tr>
<td>3</td>
<td>9 Sep</td>
<td>3 Aug</td>
<td>Feature 3</td>
</tr>
<tr>
<td>4</td>
<td>4 Oct</td>
<td>9 Sep</td>
<td>Feature 4</td>
</tr>
<tr>
<td>5</td>
<td>2 Nov</td>
<td></td>
<td>Feature 5</td>
</tr>
<tr>
<td>6</td>
<td>7 Jan</td>
<td></td>
<td>Feature 6</td>
</tr>
<tr>
<td>7</td>
<td>8 Mar</td>
<td>8 Jan</td>
<td>Feature 7</td>
</tr>
<tr>
<td>8</td>
<td>5 Apr</td>
<td></td>
<td>Feature 8</td>
</tr>
<tr>
<td>9</td>
<td>9 Apr</td>
<td></td>
<td>Feature 9</td>
</tr>
<tr>
<td>10</td>
<td>28 May</td>
<td></td>
<td>Feature 10</td>
</tr>
<tr>
<td>11</td>
<td>11 Jun</td>
<td>9 Apr</td>
<td>Feature 11</td>
</tr>
<tr>
<td>12</td>
<td>2 Jul</td>
<td></td>
<td>Feature 12</td>
</tr>
<tr>
<td>13</td>
<td>27 Jul</td>
<td></td>
<td>Feature 13</td>
</tr>
<tr>
<td>14</td>
<td>29 Oct</td>
<td></td>
<td>Feature 14</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td>Feature 15</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td>Feature 16</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td>Feature 17</td>
</tr>
<tr>
<td>18</td>
<td>8 Nov</td>
<td>20 Jul</td>
<td>Feature 18</td>
</tr>
</tbody>
</table>

Notes: Starting May 1st 2013. All dates first > 85th Confidence Interval
How I Quantify Lead Time Reduction

<table>
<thead>
<tr>
<th>Lead Time</th>
<th># Stories / Year</th>
<th>Throughput Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current</td>
<td>6</td>
<td>0%</td>
</tr>
<tr>
<td>10% Decrease</td>
<td>7</td>
<td>17% More</td>
</tr>
<tr>
<td>20% Decrease</td>
<td>8</td>
<td>33% More</td>
</tr>
</tbody>
</table>

When no ROI is easily discerned (maintenance teams?)

Even a small decrease in Cycle Time has a increased impact on throughput over a year.
# How I Quantify Cycle Time Reduction

<table>
<thead>
<tr>
<th>Cycle Time</th>
<th>Forecast Date</th>
<th>Forecast Cost</th>
<th>Cash flow to EOY14</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current</td>
<td>15-Jul-2014</td>
<td>$1,000,000</td>
<td>$0 + $60,000 = $60,000</td>
<td>0%</td>
</tr>
<tr>
<td>10% Decrease</td>
<td>27-May-2014</td>
<td>$912,500</td>
<td>$87,500 + $90,000 = $177,500</td>
<td>296% Better</td>
</tr>
<tr>
<td>20% Decrease</td>
<td>04-Apr-2014</td>
<td>$820,000</td>
<td>$120,000 + $145,000 = $265,000</td>
<td>442% Better</td>
</tr>
</tbody>
</table>

Revenue Estimates for product:

<table>
<thead>
<tr>
<th>Month</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>April</td>
<td>$30,000</td>
</tr>
<tr>
<td>May</td>
<td>$25,000</td>
</tr>
<tr>
<td>June</td>
<td>$20,000</td>
</tr>
<tr>
<td>July</td>
<td>$20,000</td>
</tr>
<tr>
<td>Aug-Dec</td>
<td>$10,000</td>
</tr>
</tbody>
</table>

Even a small decrease in Cycle Time can have huge benefit on cash flow over time.

This example shows the cost of missing a seasonal uptick in summer sales (revenue estimates shown to the left). Cost per work day is calculated at $2,500 day.
About Current Estimation Practices
Story Points and Velocity are the most common model used for forecasting software projects.

Q. Is this an accurate model?
Total Story Lead Time

Development Time
5 Days (~ 15%)

Blocked and Waiting Time
9 Days

Testing Time
2 Days

Waiting Time
3 Days

Defect Rework
2 Days

Waiting Time
8 Days

Release / DevOps Time
1 Day
Total Story Lead Time: 30 days

Development Time: 5 Days (~15%)

Blocked and Waiting Time: 9 Days

Testing Time: 2 Days

Waiting Time: 3 Days

Defect Rework: 2 Days

Waiting Time: 8 Days

Release / DevOps Time: 1 Day
Total Story Lead Time: 30 days

Pre Work: 30 days

Story / Feature Inception: 5 Days

Waiting in Backlog: 25 days

“Active Development”: 30 days

Post Work: 10 days

Waiting for Release Window: 5 Days

System Regression Testing & Staging: 5 Days
Pre Work 30 days

Total Story Lead Time 30 days

Post Work 10 days

9 days (70 total) approx 13%  

Discuss Survey Question: “Does card size matter?”

Story / Feature Inception 5 Days

Waiting in Backlog 25 days

“Active Development” 30 days

Waiting for Release Window 5 Days

System Regression Testing & Staging 5 Days
Q. Do story point estimates correlate to story cycle-time or lead-time?

• Answer: No. Unless...
  – You can remove the pre-tasks (small stories, JIT design)
  – You can remove the post-tasks (cont. deployment / testing)
  – You can reduce the wait time between tasks (staff balance)

• Story effort accounts for < 15% cycle time

• For velocity to be a good predictor, some correlation between points and the unknown wait times would need to be present - It’s NOT
  – Story size is in NO way related to 52 of the 70 days in our example (72%)

Discuss Survey Question: “How does Velocity and Story Points compare?”
Exercise – 5 Minutes

• Describe the story development to release cycle time budget in your companies
• Discuss and describe the likely ranges and uncertainty of the major steps
Using KanbanSim and ScrumSim
http://bit.ly/.....

1. Download the zip file
2. Un-compress to a location of your choice
3. Double-click on the .exe file
Launching & Installing

• No installation – just unzip
• Requires the Microsoft .NET Framework 4 or later to execute; installed on most machines
• Folder structure is important (and editable)
  – Examples and Resources
  – Naming convention
• Launch by double clicking the .exe file
Running the Examples

Click to open

Drag to move timeline
Editing / Debugging the Model

- Editor
- Tabs
- Line Numbers
- Group Collapse
- Color Highlighting
Charts and Statistics

- Cycle time
- Blocked Positions
- Active Positions
- Empty Positions
- Queued Positions
- Active / Inactive Positions
- Intervals
Forecasts

Delay based on targetDate and revenue entries

Yellow indicated first entry > targetLikelihood
Exercise – First Monte Carlo Sim
15 Minutes

• Start KanbanSim and open the example file
  1 - Lead Time Forecast (uniform)
• Explore the ribbon toolbar options (follow me)
• Explore the different charts (follow me)
• Follow the exercises text within the model
  – Monte Carlo shape and Confidence Intervals
  – Forecasting Dates, Cost and Cost of Delay
Exercise – Multiple Deliverables
10 Minutes

• Group discussion
• Start KanbanSim and open the example file
  1.1 - Multiple Deliverable Lead Time Forecast (uniform)
• Follow the exercises text within the model
  – Creating multiple backlogs
  – Controlling Order
Exercise – Accounting for Scope Creep
10 Minutes

• Discuss how to define scope creep
• Start KanbanSim and open the example file 8 - Scope Creep Lead Time Forecast
• Follow the exercises text within the model
Accounting for Risk
Flaw of Averages

50% Possible Outcomes

50% Possible Outcomes
We need to estimate risk events

**Major risk events have the predominate role in deciding where deliver actually occurs**

We spend all our time estimating here
Risk likelihood changes constantly

95th Confidence Interval

1 2 3
Risk likelihood changes constantly

95th Confidence Interval

1

2

3

Commercial in confidence
Risk likelihood changes constantly

95th
Confidence Interval

1

2

3

Commercial in confidence
Risk likelihood changes constantly

Confidence Interval

95\textsuperscript{th}

1  2  3

Commercial in confidence
Exercise – Account for Risk
10 Minutes

• Group discussion
• Start KanbanSim and open the example file
  1.2 - Risk Lead Time Forecast (uniform)
• Follow the exercises text within the model
  – Accounting for risks and conditional scope
Introduction to Cost of Delay
Cost of Delay to Make Decisions on More Than Development Cost
Eg. Not hiring a tester will cost us $x per day

Cost of Delay for Prioritization
Eg. Which order should features/projects be started to maximize cashflow

Cost of Delay as a Unit of Currency
Eg. Company wide currency for decision making
Cost of Delay Resources

• Don Reinertsen

• Joshua Arnold – good practical application
  – http://costofdelay.com
Cost of Delay

• Cost of Delay is the right metric to prioritize work

• Consists of the sum of –
  – Missed Revenue
    • New revenue
    • Protecting Existing revenue
  – Extra costs
    • Running costs (dev staff, ops staff, ops cost)
    • Fines and penalties (regularity, contractual)
  – Sum of Cost of Delay of Other Projects (often missed)
Forecast Completion Date

Planned / Due Date  Actual Date

Cost to Develop ($)

Dev Cost of Delay ($$$)

NOT total cost of delay, just extra costs

July  August  September  October  November  December
Reinertsen’s WSJF prioritization algorithm does a good job of highlighting the issues.

See the spreadsheet [Cost of Delay.xlsx](#)
Cost of Delay by Duration

- Cost of Delay has to be in the same unit of duration to compare and prioritize
- Cost of Delay by Duration
  
  \[ CD3 = \frac{\text{Cost of Delay}}{\text{Duration}} \]

**Cost of Delay Divided by Duration** (CD3) is a scheduling method that improves how we prioritize projects and features. CD3 is useful in situations where there is limited capacity to deliver – a situation which is common in product development, especially software development. CD3 is effectively a form of the “Weighted Shortest Job First” queuing method. In this case we are weighting by **Cost of Delay**. When using CD3, priority order of features or projects is determined by dividing the estimated Cost of Delay by the estimated duration: the higher the resulting score, the higher the priority.” by CostOfDelay.com
Lifetime Value

• Features and products have a lifecycle
• Different market pressures apply
  – Growth Rate: Value growth to peak over time
    • Upgrade base
    • New market entry (early adopters, but slow growth)
  – Decay Rate: Value decay rate over time
    • Long term and short term
  – Competitiveness: Late entry erodes peak value
    • Lose customers to competitors
    • Date dependent: No value if after a certain date
Long Term Opportunity, with Fixed End Date

Cost of Delay

Short Term Opportunity, with Fixed End Date

Sometimes no peak erosion
Questions...

• What date does it no longer make sense to release
• Will this generate revenue (how much)
  – How quickly will adoption occur
    • % Earliest adopters, % general, % laggards, never
  – Will this gain market share (how much)
    • How quickly will competitors neutralize
      – They know the idea too, it’s a race
      – When will they find out what we are doing (apple)
    – Have we set a price
• Will this protect revenue (how much)
  – If we don’t do this, what is the loss of revenue and market share
• Will this reduce costs (how much)
• Will we incur fines or penalties at a certain date
  – Are we compelled to do this feature
  – Should we do this anyway (good citizens / PR)
• What other features are impacted by this feature being late (staff, etc.)
Exercise – 10 Minutes

- Draw the life cycle value over time curves for:
  - Short term opportunity
  - Long term opportunity
- What is the impact of being late?
  - On lifetime value
  - How would strong competition erode value
- Discuss your produces and market conditions
Always Ask

When do you need it?

When cost of delay is high, it's important to find an option that achieves that goal, often at GREAT cost.
Forecast Completion Date

Cost to Develop

Staff A: $$$$$$*
Staff B: $$
Staff C: $

Planned / Due Date

Actual Date A
Actual Date B
Actual Date C

Staff Option A
Staff Option B
Staff Option C

July August September October November December
Shape and Impact of Cycle-Time & Scope
Note: Histogram from actual data
Probability Density Function

Histogram Weibull

$0.32$
$0.28$
$0.24$
$0.20$
$0.16$
$0.12$
$0.08$
$0.04$
$0.00$

$-10$
$0$
$10$
$20$
$30$
$40$
$50$
$60$
$70$
$80$
$90$
$100$
$110$
$120$

$0.32$
$0.28$
$0.24$
$0.20$
$0.16$
$0.12$
$0.08$
$0.04$
$0.00$

Histogram

Weibull
Why Weibull

- Now for some Math – I know, I’m excited too!

- Simple Model
- All units of work between 1 and 3 days
- A unit of work can be a task, story, feature, project
- Base Scope of 50 units of work – Always Normal
- 5 Delays / Risks, each with
  - 25% Likelihood of occurring
  - 10 units of work (same as 20% scope increase each)
Normal, or it will be after a few thousand more simulations.
5th %: 62  25th%: 63  75th%: 72  95th%: 78

Base + 1 Delay

Histogram

Count

Up to and including values for Intervals (Monte Carlo)
Base + 2 Delays
5th %: 62  25th%: 65  75th%: 78  95th%: 91

Histogram

Base + 3 Delays

Count

61.45  66.35  71.25  76.15  81.05  85.95  90.85  95.75  100.65  105.55

Up to and including values for Intervals (Monte Carlo)
Histogram

Base + 4 Delays

Sample Count: 5000  Min: 59  Avg: 76.941  Median: 77  Max: 118  Standard Dev: 10.766

5th %: 62  25th%: 66  75th%: 86  95th%: 93
5th %: 63  25th%: 74  75th%: 89  95th%: 103

Histogram

Base + 5 Delays

Count: 785
<backlog type="custom">
  <deliverable name="Base">
    <custom count="50" />
  </deliverable>
  <deliverable name="Delay1" skipPercentage="75">
    <custom count="10" />
  </deliverable>
  <deliverable name="Delay2" skipPercentage="75">
    <custom count="10" />
  </deliverable>
  <deliverable name="Delay3" skipPercentage="75">
    <custom count="10" />
  </deliverable>
  <deliverable name="Delay4" skipPercentage="75">
    <custom count="10" />
  </deliverable>
  <deliverable name="Delay5" skipPercentage="75">
    <custom count="10" />
  </deliverable>
</backlog>

<columns>
  <column id="1" estimateLowBound="1" estimateHighBound="3" wipLimit="2">Work</column>
</columns>

<forecastDate startDate="01-May-2012" costPerDay="2500" />
| Mining Cycle Time Data |
Pattern: Total Cycle Time

- Story / Feature Inception: 5 Days
- Waiting in Backlog: 25 Days
- “Active Development”: 30 Days
- Waiting for Release Window: 5 Days
- System Regression Testing & Staging: 5 Days
Pattern: Segmented Cycle Time

- **Lead Time**
  - Story / Feature Inception: 5 Days
  - Waiting in Backlog: 25 Days

- **Dev Cycle Time**
  - “Active Development”: 30 Days

- **Test Cycle Time**
  - Waiting for Release Window: 5 Days
  - System Regression Testing & Staging: 5 Days
Mining / Testing Cycle Time Data

Capture Data

Fit Shape

Histogram

Scatter Plot
Cycle Time Capture Practices

• Clearly understand from where and to where
• Capture begin and end date; compute the number of days in-between
• Does cycle time include defect fixing?
• Are there multiple types of work in the same cycle time data
  – Stories
  – Defects
  – Classes of Service
Things that go wrong...

• Zero values
• Repetitive (erroneous) values
• Batching of updates
• Include/exclude weekends
• Project team A staff raided impacting their cycle time (Team A up, team B’s down)
• Work complexity changes
• Team skill changes
Many low values. Often zero or values below what makes sense. Check the most frequent low values.

Multiple modes. In this case two overlapping Weibulls. Often due to multiple classes of service, or most often - defects versus stories.
Probability Density Function

Histogram

Weibull

\[ \begin{align*}
\alpha &= 1.5178 \\
\beta &= 31.965 \\
\gamma &= 0
\end{align*} \]

Shape – How Fat the distribution.

Scale – How Wide in Range. Related to the Upper Bound

Location – The Lower Bound
Estimating Distributions using Historical Data
What Distribution To Use...

• No Data at All, or Less than < 11 Samples *(why 11?)*
  – Uniform Range with Boundaries Guessed (safest)
  – Weibull Range with Boundaries Guessed (likely)
• 11 to 50 Samples
  – Uniform Range with Boundaries at 5\(^{th}\) and 95\(^{th}\) CI
  – Weibull Range with Boundaries at 5\(^{th}\) and 95\(^{th}\) CI
  – Bootstapping (Random Sampling with Replacement)
• More than 100 Samples
  – Use historical data at random without replacement
  – Curve Fitting

Discuss Survey Question: “Brand new team, can I forecast?”
Tools

- EasyFit from [http://mathwave.com](http://mathwave.com) $499
- R [http://cran.r-project.org](http://cran.r-project.org) Free
- Statistics feature of KanbanSim [http://focusedobjective.com](http://focusedobjective.com) Free
- Excel of course
- Monte Carlo features of KanbanSim $495-995
- Google for Monte Carlo Simulation Tools
  - Oracle, Palisade, HubbardResearch (he may be here, so a big recommendation!)
Sampling at Random Strategies

• If you pick what samples to use, you bias the prediction...

• Strategies for proper random sampling –
  – Use something you know is random (dice, darts)
  – Pick two groups using your chosen technique and compute your prediction separately and compare
  – Don’t pre-filter to remove “outliers”
  – Don’t sort the data, in fact randomize more if possible
Estimating Concurrent Effort from Cumulative Flow Chart

Concurrent WIP Sample: Find the smallest and the biggest or take at least 11 samples to be 90% sure of range.
Scope Creep Over Time

Look at the rate new scope is added over time
Historical Data
Forecasting
Exercise – Built-In Weibull
10 Minutes

• Discuss how to estimate Weibull curve
• Start KanbanSim and open the example file 2 - Lead Time Forecast (weibull)
• Follow the exercises text within the model
Exercise – Sample Data
10 Minutes

- Discuss how to gather sample data
- Start KanbanSim and open the example file 3 - Lead Time Forecast (sample data)
- Follow the exercises text within the model
Exercise – Valuing 10% Improvement
20 Minutes

• Discuss how to calculate 10% Cycle Time improvement by throughput and cashflow
• Start KanbanSim and open the example file 4 - 10 Percent Improvement
• Follow the exercises text within the model
Exercise – Multiple Project Phases
15 Minutes

• Discuss how to define and use multiple project phases in a model
• Start KanbanSim and open the example file 5 - Multiple Phase Lead Time Forecast
• Follow the exercises text within the model
Accounting for Skill and Role Impact
1. Historical Cycle Time

2. Planned Resources/Effort

3. The Work (Backlog)

4. Historical Scope Creep Rate

5. Historical Defect Rate & Cycle Times

A Process to Combine Multiple Uncertain Measurements/Estimates is Needed
Exercise – Multiple Role / Staff
15 Minutes

• Discuss how to assess what staff role has the greatest impact and how to set effort/wip

• Start KanbanSim and open the example file 6 - Multiple Role Lead Time Forecast (uniform)

• Follow the exercises text within the model
Exercise – Targeted Effort by Skill
15 Minutes

• Discuss how to define a backlog that emphasize certain staff skills
• Start KanbanSim and open the example file 7 - Expertise Emphasis - Multiple Role Lead Time Forecast (uniform)
• Follow the exercises text within the model
Exercise: 30 Minutes

- Open the Kanban\Typical Board example
- Perform a forecast (write down the result)
- Perform Staff analysis and determine what staff additions make the most impact
- Make this change in the model and re-forecast
- Perform a sensitivity analysis and determine the impact of reducing defect occurrence rates
What you will learn....
(in order of survey responses)

• How to forecast likely delivery dates and costs
• How to calculate Cost of Delay and why it matters
• How to quantify the benefits of decreasing cycle time in cash-flow and throughput
• How to calculate the cost and date impact of defects and blocking events
• How to analyze systems for staff imbalance
• How to analyze and use historical data when available
• What constitutes "enough" historical data and what to do when you have less (or none)
What topics or questions did we miss. List three more topic areas

• Story points and velocity (Scrum) relationships
• What the prerequisites to getting a believable delivery schedule
• (Automated) Tools that help with calculations and predictions
• Card size -- do they all have to be the same?
• Little's law and WIP and its relationship with prediction
• I'm creating a new team for a new project, am I screwed when my boss asks for a schedule?
Any general requests or questions you have

- Please give us a lot of notes to take away. Tools that we can use. And a contact email that we can validate our learnings without fear.
- Monte Carlo analysis is great, but can you present other statistical/quantitative techniques to base predictions on?
- I'm really interested in the technique, but equally important is the hard part: Having the conversations with the right folks in the right way to get them to use the math responsibly and accurately. I'd love help with both!
Conference Special: Download the session slides, a free copy of our simulation software and a copy of this book in PDF format from http://bit.ly/agilesim

Focused Objective software risk solutions
FocusedObjective.com
@AgileSimulation

Troy Magennis
troy.magennis@focusedobjective.com
phone: 425 223 8097  skype: troy.magennis  twitter: @t_magennis
Data Shape, Sampling and Prediction Intervals
What are the chances the next sample will be within the range of previous samples

SAMPLING AND PREDICTION INTERVALS
Q. What is the chance of the 4\textsuperscript{th} sample being between the range seen after the first three samples?
(no duplicates, uniform distribution, picked at random)
Q. What is the chance of the 4th sample being between the range seen after the first three samples?
(no duplicates, uniform distribution, picked at random)
Q. What is the chance of the 4th sample being between the range seen after the first three samples?
(no duplicates, uniform distribution, picked at random)

A. 50%

\[ % = \left(1 - \frac{1}{n - 1}\right) \times 100 \]
Q. What is the chance of the 12th sample being between the range seen after the first three samples?
(no duplicates, uniform distribution, picked at random)

A. 90%

\[
\% = (1 - \left(\frac{1}{n - 1}\right)) \times 100
\]
<table>
<thead>
<tr>
<th># Prior Samples</th>
<th>Prediction Next Sample Within Prior Sample Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>50%</td>
</tr>
<tr>
<td>4</td>
<td>67%</td>
</tr>
<tr>
<td>5</td>
<td>75%</td>
</tr>
<tr>
<td>6</td>
<td>80%</td>
</tr>
<tr>
<td>7</td>
<td>83%</td>
</tr>
<tr>
<td>8</td>
<td>86%</td>
</tr>
<tr>
<td>9</td>
<td>88%</td>
</tr>
<tr>
<td>10</td>
<td>89%</td>
</tr>
<tr>
<td>11</td>
<td>90%</td>
</tr>
<tr>
<td>12</td>
<td>91%</td>
</tr>
<tr>
<td>13</td>
<td>92%</td>
</tr>
<tr>
<td>15</td>
<td>93%</td>
</tr>
<tr>
<td>17</td>
<td>94%</td>
</tr>
<tr>
<td>20</td>
<td>95%</td>
</tr>
</tbody>
</table>
Exercise – 10 Minutes

- Throw a 10 sided dice + 10 sided % dice; total
- Record the values (0 to 99) of 25 throws

<table>
<thead>
<tr>
<th>Value</th>
<th>Lowest So Far</th>
<th>Highest So Far</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Discuss...
Sampling at Random Strategies

- If you pick what samples to use, you bias the prediction...

- Strategies for proper random sampling –
  - Use something you know is random (dice, darts)
  - Pick two groups using your chosen technique and compute your prediction separately and compare
  - Don’t pre-filter to remove “outliers”
  - Don’t sort the data, in fact randomize more if possible
BEST PRACTICES
The Model Creation Cycle

- Sensitivity Test
- Model (a little)
- Monte-Carlo Test
- Visually Test
The Experiment Cycle

Baseline

Make Single Change

Make Informed Decision(s)

Compare Results
Best Practice 1

Start simple and add ONE input condition at a time.

Visually / Monte-carlo test each input to verify it works
Best Practice 2

Find the likelihood of major events and estimate delay
E.g. vendor dependencies, performance/memory issues, third party component failures.
Best Practice 3

Only obtain and add detailed estimates and opinion to a model if Sensitivity Analysis says that input is material
Best Practice 4

Use a uniform random input distribution UNTIL sensitivity analysis says that input is influencing the output.
Best Practice 5

Educate your managers’ about risk. They will still want a “single” date for planning, but let them decide $75^{th}$ or $95^{th}$ confidence level (average is NEVER an option)
Best Practice 6

Compare Multiple Models to Find Errors and to Answer Different Questions. For example, compare velocity based model against cycle time model.
Kanban Backlog Model Granularity

1. Simple Backlog (# stories)

2. Multiple backlog entries (# stories)

3. Multiple entries with column cycle-time percentage overrides

4. Multiple entries with column cycle-time estimate overrides

5. Every story listed individually with column percentage or estimate overrides
1. Simple Backlog

...<setup>
  <backlog type="simple" simpleCount="10" />
...  

• Primes the initial backlog with 10 cards
• Cards use the cycle-time estimates defined for each column
• Useful for testing the model before breaking out the backlog into more detail
2. Multiple Backlog Entries

<setup>
  <backlog type="custom" shuffle="true">
    <custom name="Small" count="6" />
    <custom name="Medium" count="4" />
  </backlog>
</setup>

- Primes the initial backlog with 10 cards
- Kanban: Cards use the cycle-time estimates defined for each column
3. Percentage Override

<setup>
  <backlog type="custom" shuffle="true">
    <custom name="Small" count="6"
      percentageLowBound="0" percentageHighBound="66" />
    <custom name="Medium" count="4"
      percentageLowBound="33" percentageHighBound="100" />
  </backlog>
</setup>

• Primes the initial backlog with 10 cards
• Cards use the percentage range of the cycle-time estimates defined for each column
• Great for rapid modeling of rough order of magnitude estimates
4. Column Override

<setup>
<backlog type="custom" shuffle="false">
  <!-- override the first two columns, the inherit all others -->
  <custom name="small" count="10">
    <column id="1" estimateLowBound="1" estimateHighBound="3" />
    <column id="2" estimateLowBound="1" estimateHighBound="3" />
  </custom>
</backlog>

• Primes the initial backlog with 10 cards
• Cards use the specific column cycle-time if listed
• Cards use the percentage range of the cycle-time estimates defined for each column not listed
Deliverable Model Granularity

None defined

One defined

Multiple deliverables defined
Scrum Backlog Model Granularity

- Single Backlog entry (# stories)
- Multiple backlog entries (# stories) with specific estimate
- Every story listed individually with specific estimate
Estimation Model Granularity

- Low bound and high bound uniform
- Built-in distribution curves (20+)
- Custom distribution (histogram guessed)
- Custom distribution (from historical data)
- Markov chains from historical data
Project Phases Granularity

- No phases defined
- Named phases defined
- Phases defined with global multiplier(s)
- Events target specific phases
- Phases override column WIP limits
Kanban Defect Event Model
Granularity

1. No Defects defined
2. One defect defined
3. Multiple types of defects defined
4. Defects target specific phases
5. Defects override column estimates
Scrum Defect Event Model
Granularity

1. No Defects defined
2. One defect defined
3. Multiple types of defects defined with specific estimates
4. Defects target specific phases
Added Scope Model Granularity

- No Added Scopes defined
- One added scope defined
- Multiple types of added scopes defined
- Added scopes target specific phases
Blocking Event Model Granularity

No Blocking Events defined

One blocking event defined

Multiple types of blocking events defined

Multiple types of blocking events targeting specific phases
Model Patterns
Modeling Software is Complex

- Pre Conditions
  - Start Date
- Scope
  - Known work
  - Defect work
  - Scope Additions
  - Scope Change
  - Risks
- Parallel effort
  - Staff size and onboarding strategy
  - Skill Constraints and Imbalances
- Uncertainty
  - All estimates (actually, all numbers)
  - Market conditions
- Process
  - Development Cadence
  - Release Cadence
  - Location / Co-location
  - Wait/Queued Time
  - Lead Time
  - Blocked Time
- ...
Model

Scope

Process

Effort
# Model Patterns - Process

<table>
<thead>
<tr>
<th>Cycle Time</th>
<th>Velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Status states</strong></td>
<td><strong>Backlog Points</strong></td>
</tr>
<tr>
<td><strong>Status cycle time</strong></td>
<td><strong>Velocity</strong></td>
</tr>
<tr>
<td><strong>Status WIP limits</strong></td>
<td><strong>By Phase adjustment</strong></td>
</tr>
<tr>
<td><strong>By Phase adjustment</strong></td>
<td></td>
</tr>
<tr>
<td><strong>By Type of work</strong></td>
<td></td>
</tr>
</tbody>
</table>
### Sources of Scope

<table>
<thead>
<tr>
<th>Initial Scope</th>
<th>New Scope</th>
<th>Impediments</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Deliverables</td>
<td>• Creep</td>
<td>• Blockers</td>
</tr>
<tr>
<td>• Features</td>
<td>• Change</td>
<td>• Queue Time</td>
</tr>
<tr>
<td>• <strong>Stories</strong></td>
<td>• Defects</td>
<td>• Friction</td>
</tr>
</tbody>
</table>

Commercial in confidence
**Risk events have a pivotal role in delivery**

Whilst “Bell Curve” makes for easy explanation, project release probability is more complex based on Risks and Dependencies. This needs to be considered. (covered later in this presentation)
Model Patterns - Effort

Total Cycle Time

Segmented Cycle Time

Step by Step Process
Pattern: Total Cycle Time

- Story / Feature Inception: 5 Days
- Waiting in Backlog: 25 Days
- “Active Development”: 30 Days
- Waiting for Release Window: 5 Days
- System Regression Testing & Staging: 5 Days
Pattern: Segmented Cycle Time

- **Story / Feature Inception**: 5 Days
- **Waiting in Backlog**: 25 Days
- **“Active Development”**: 30 Days
- **Waiting for Release Window**: 5 Days
- **System Regression Testing & Staging**: 5 Days
Velocity is NOT Linear

nor is defect rate, scope-creep, story expertise requirements, team skill, etc.
Distributions and Historical Data
# Distribution Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The text name of this distribution. Used in distribution=&quot;&quot; strings.</td>
</tr>
<tr>
<td>shape</td>
<td>The built-in distribution shape strings.</td>
</tr>
<tr>
<td>generator</td>
<td>The random number generator to use.</td>
</tr>
<tr>
<td>parameters</td>
<td>The comma parameters for this distribution. Different for each shape.</td>
</tr>
<tr>
<td>lowBound</td>
<td>The lowest value allowed, scaled using the rangeProcessing attribute.</td>
</tr>
<tr>
<td>highBound</td>
<td>The highest value allowed, scaled using the rangeProcessing attribute.</td>
</tr>
<tr>
<td>rangeProcessing</td>
<td><em>clip</em> or <em>stretch</em>. Clip omits values outside of the range, stretch fits the random numbers from the distribution to the width of lowBound and highBound.</td>
</tr>
<tr>
<td>location</td>
<td>Where “0” from the distribution starts on the X-Axis.</td>
</tr>
<tr>
<td>numberType</td>
<td><em>double</em> or integer.</td>
</tr>
<tr>
<td>separatorCharacter</td>
<td>For distributions that use sample data, this is the separator character.</td>
</tr>
<tr>
<td>count</td>
<td>The initial number of random samples to produce.</td>
</tr>
<tr>
<td>zeroHandling</td>
<td><em>keep</em>, remove, or value. If value is chosen, zeroValue is used for zero’s.</td>
</tr>
<tr>
<td>zeroValue</td>
<td>The value to replace zero’s in the sample data when ‘value’ is specified.</td>
</tr>
</tbody>
</table>
fromSamples Distribution

- Bootstraps historical samples into a larger set of random numbers
- Handles 0’s in specific fashions
- Samples separators can be comma, pipe or newline
- comma cannot be used as a separator for cultures that use , as a decimal separator

```xml
<distribution name="cycleTimesTesting" shape="fromSamples" count="1000" zeroHandling="value" zeroValue="0.5">0 17 1 22 0 22 3 3 ...
</distribution>
```