Bummler und Schummler – wie effizient ist mein UI wirklich?

Bearbeitungszeiten analysieren und verstehen mit Probability Plots

Bernard Rummel
SAP AG
Dietmar-Hopp-Allee 16
69190 Walldorf
bernard.rummel@sap.com

Abstract
Bearbeitungszeiten gehören zum klassischen Instrumentarium von Usability Tests, um die Effizienz des UIs zu erfassen. Was nach objektiver Messung aussieht, hat aber einige Tücken. Was kann man tun, wenn nicht alle Benutzer die gestellte Aufgabe lösen? Rechnet man nur die erfolgreichen Benutzer ein – „survival of the fittest" – erscheint das UI effizienter, als es in Wirklichkeit ist. Auch ein besonders schneller user könnte, besonders in unmoderierten Tests, immer auch geschummelt haben. Da Zeiten selten normalverteilt sind, geben Mittelwert und Standardabweichung ein schiefes Bild – überlange Zeiten sind oft keine Ausreißer, sondern statistisch zu erwarten.
Probability Plotting, eine graphische Methode aus der technischen Zuverlässigkeitsanalyse, löst diese Probleme auf elegante Weise. Die Betrachtung verschiedener Verteilungstypen ermöglicht neue Einsichten – z.B. die getrennte Betrachtung von technischer Performance und Effizienz des UI-Designs. Die Plots erlauben ein schnelles Screening der Daten auf Auffälligkeiten; damit eignet sich die Methode besonders für unmoderierte Online-Tests.

Keywords:
/// Usability-Testing
/// Bearbeitungszeit
/// Effizienz
/// Analysemethoden

1. Effizienz: „State of The Art" – und Probleme damit

Aufgaben-Bearbeitungszeiten werden in Usability-Tests typischerweise gemessen, um die Effizienz des UIs zu erfassen. Nach ISO9241–11:1998 beschreibt Effizienz eigentlich den **Aufwand**, den ein User zur Erreichung seiner Ziele betreiben muss, in Relation zur Genauigkeit und Vollständigkeit der Lösung – gemeint ist also nicht nur der **zeitliche** Aufwand. Trotzdem hat sich die Zeitmessung als „offensichtlich" objektive Metrik soweit etabliert, dass sie auch im **Common Industry Format for usability test reports** (CIF, ISO/IEC 25062) explizit eingefordert wird. Üblich und standardkonform ist dabei die Angabe von Mittelwert, Standardabweichung, Minimum und Maximum der Bearbeitungszeit pro Aufgabe.

Was vernünftig und objektiv klingt, ist in der Praxis gar nicht so einfach:

1. Welche Daten sollen überhaupt in die Analyse eingehen?
   Was ist mit Personen, die die gestellte Aufgabe nicht lösen? Schließt man sie aus, überschätzt man die Effizienz des UIs. Aber was wäre die Alternative?
   Was ist mit Bummlern und Schummlern, also Personen, die ganz oder teilweise etwas anderes tun, als die Aufgabe zu bearbeiten? Wie kann man überhaupt „Ausreißer" entdecken, die ungewöhnlich lange brauchen oder auffällig schnell sind – wann ist eine Bearbeitungszeit nicht mehr „normal"?
2. Was ist mit unterschiedlichen Systemantwortzeiten?
   Technische Antwortzeiten können einen mehr oder weniger großen Anteil der Bearbeitungszeit ausmachen. Wollen wir Vergleiche z.B. zwischen verschiedenen mobilen Plattformen anstellen, müssen wir technische Einflüsse von Designproblemen sauber trennen. Gerade Smartphones lassen sich aber nicht so einfach für entsprechende Zeiterfassungen instrumentieren – und mit der nächsten Gerätegeneration oder OS-Version sähe wieder alles anders aus.
3. Sind Mittelwert und Standardabweichung überhaupt geeignete Metriken?
   In der Literatur findet man immer wieder Hinweise, dass Bearbeitungszeiten nicht normalverteilt seien (z.B. Sauro & Lewis, 2009, Sauro, 2011; ausführliche Behandlung bei Luce, 1986), kurioserweise aber kaum Angaben, welche Verteilungen denn nun vorliegen. Dabei kann diese Information entscheidend sein: bei Exponential- oder Lognormalverteilungen sind überlange Bearbeitungszeiten

keine Ausnahmen, sondern statistisch zu erwarten. Verwendet man statt des Mittelwerts Median oder geometrisches Mittel (wie z.B. Sauro, 2011 empfiehlt), erscheinen solche Zeiten noch exotischer, als sie sind, da das geometrische Mittel typischerweise kleiner als das arithmetische Mittel ist und der Unterschied dadurch noch krasser ausfällt.

4. Wie kann man Effizienz überhaupt parametrisieren?
   Natürlich hängt die Bearbeitungszeit für eine Aufgabe auch von deren Komplexität ab. Ein effizientes Design zeichnet sich dadurch aus, dass gerade komplexe Aufgaben schnell erledigt werden können. Wie kann man aber die Effizienz von Designs bei verschiedenen Aufgaben vergleichbar erfassen? Sauro (2011) schlägt verschiedene Strategien vor, die letztlich auf Vergleich mit „kritischen" Bearbeitungszeiten beruhen. Solche Zeiten können z.B. maximal akzeptable Prozesszeiten, Bearbeitungszeiten von vergleichbaren Produkten, Vielfache von „Ideal"- oder „Experten"-Zeiten usw. sein. Die „bootstrapped specification limit"-Zeit (Sauro & Kindlund, 2005) ist die **maximale** Zeit, die Benutzer mit einer **minimalen** definierten Benutzerzufriedenheit brauchten („Wie lange darf man brauchen, um nicht gar zu unzufrieden zu sein?"). All diese Ansätze sind insofern problematisch, als sich Messfehler und zufällige Streuungen in der Zeitmessung durch den Vergleich erheblich aufaddieren können.

2. Probability Plotting

Probability Plotting ist eine graphische Methode aus der technischen Zuverlässigkeitsanalyse, die einige der genannten Probleme elegant löst. Zuverlässigkeitsingenieure analysieren typischerweise Ausfallzeiten, also die Zeit bis zum Versagen eines Teils. Wir hingegen betrachten die Zeit bis zum Erfolg des Benutzers, haben also genau die umgekehrte Perspektive. Die Mathematik und einige zentrale Konzepte lassen sich trotzdem gut übertragen. Der Grundgedanke besteht darin, die beobachteten Zeiten zu sortieren und gegen diejenigen Perzentilwerte zu plotten, bei denen man sie unter Vorliegen einer bestimmten Verteilungsannahme erwarten würde. Passen die beobachteten Daten zur erwarteten Verteilung, erscheinen die Datenpunkte als gerade Linie. Ausreisserwerte sind unmittelbar daran erkennbar, dass sie von einem ansonsten systematischen Muster abweichen. Hat man die zugrundeliegende Verteilung identifiziert, kann man aus dem entsprechenden Plot Verteilungsparameter ermitteln, die durchaus informativer sein können als Mittelwerte.

2.1. Probability Plots erstellen

Das **e-Handbook of Statistical Methods** des US National Institute of Standards and Technology (NIST/SEMATECH 2012a) bietet eine frei zugängliche Schritt-für-Schritt- Beschreibung zur Erstellung von Probability Plots (NIST/SEMATECH 2012b), auf die hier daher verzichtet wird. Eine xls-Datei mit den für usability professionals wichtigsten Plots kann beim Autor angefragt werden. Die folgende Darstellung beschränkt sich auf konzeptionelle Aspekte.

Betrachten wir eine Menge von Teilen bzw. Benutzern. Zu einem gegebenen Beobachtungszeitpunkt kann es vorkommen, dass das betreffende Teil noch nicht ausgefallen ist, bzw. der betrachtete Benutzer die Aufgabe noch nicht gelöst hat. Wann genau das passieren wird, weiß man nicht; jedoch ist sicher, dass es bis zum Beobachtungszeitpunkt noch nicht passiert ist. Derartige Daten werden in der Zuverlässigkeitsanalyse als **censored** bezeichnet. Bei Aufgaben ohne Zeitlimit kommt es oft vor, dass ein Benutzer zu einem bestimmten Zeitpunkt aufgibt oder aber eine falsche Lösung angibt. Da dies für jeden Benutzer zu einem individuell verschiedenen Zeitpunkt vorkommen kann, müssen Usability-Testdaten als **multiply censored** betrachtet werden.

Das Censoring-Konzept erlaubt uns, auch die Daten nicht erfolgreicher Benutzer in die Analyse einzubeziehen. Wir wissen ja, dass ein Benutzer bis zum Feststellen des Misserfolgs die Aufgabe nicht gelöst hat, und können diese Information nutzen. Die Zeiten, zu denen Benutzer eine Aufgabe gelöst oder eben nicht gelöst haben, werden dazu einfach sortiert, zusammen mit der Information, welche Benutzer erfolgreich waren. Die Rangnummern werden dann mit der **Modified Kaplan-Meier (K-M) Product Limit procedure** (NIST/SEMATECH 2012c) korrigiert, die im Wesentlichen darin besteht, jedem Benutzer i mit seiner individuellen Bearbeitungszeit t_i einen Prozentrang $R(t_i)$ zuzuweisen, der seiner Position in einer theoretischen Gesamtpopulation aller Benutzer entspräche, und zwar inklusive der nicht erfolgreichen Benutzer. Vereinfacht gesprochen ist R(t) der Anteil Benutzer, von dem man erwartet, die Aufgabe **nicht** zu lösen. Die beobachteten Zeiten t_i werden nun gegen die Prozentränge $R(t_i)$ geplottet. Die Achsen werden dabei spezifisch für jeden Verteilungstyp spezifisch transformiert (NIST/SEMATECH 2012b):

- Für Exponentialverteilungen ist die R-Achse logarithmisch, die Zeitachse linear skaliert
- Für Weibull-Verteilungen sind beide Achsen logarithmisch skaliert
- Für Normal- und Lognormalverteilungen werden statt der Prozentränge R(t) die inversen Normalverteilungswerte des Komplements z(1-R(t)) verwendet, bei der Lognormalverteilung wird zusätzlich die Zeitachse logarithmisch skaliert.

Das Ergebnis sind im vorliegenden Fall vier Plots, jeweils einer für Exponential-, Weibull-, Normal- und Lognormalverteilung. Passen die Daten zu einer bestimmten Verteilung, erscheinen die Datenpunkte im entsprechenden Plot als gerade Linie[1]. **[Abb. 1]**

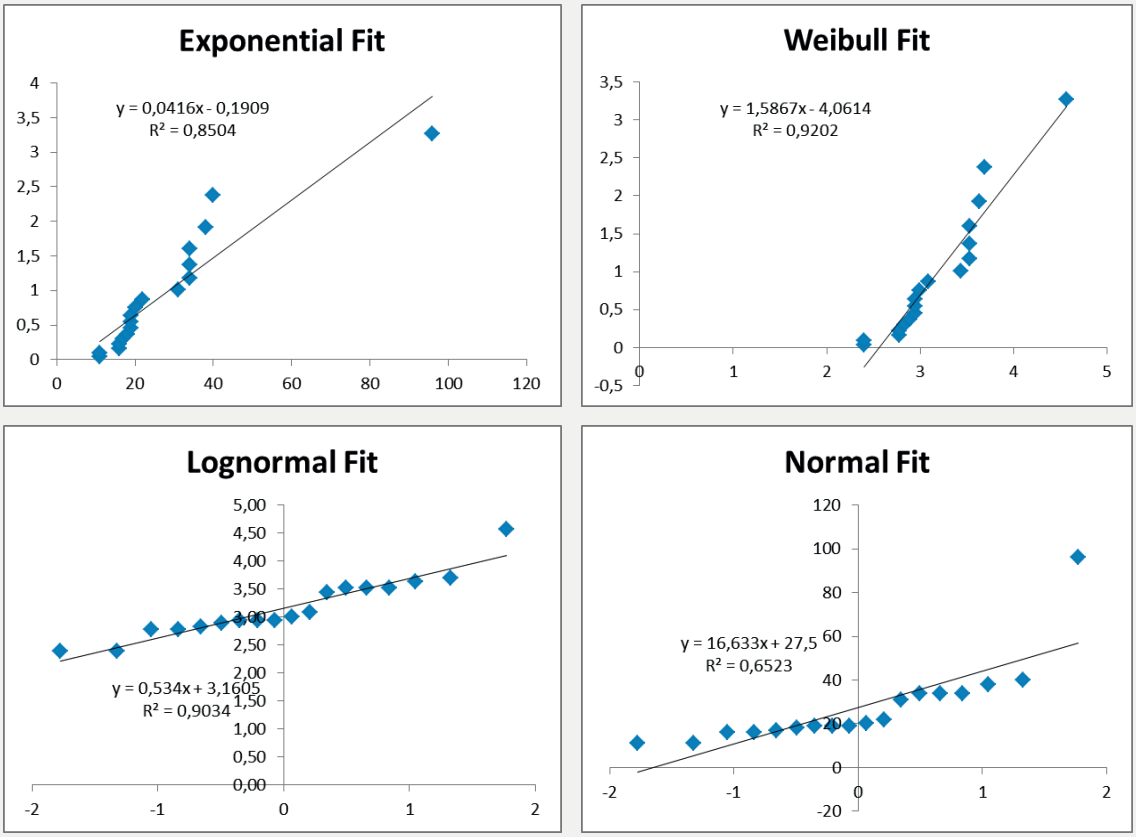

Abb. 1.
Probability Plots für Bearbeitungszeiten von 18 Benutzern einer Smartphone-Anwendung zur Reisekostenerfassung. Exponential- und Normalverteilungsplot deuten auf einen „Bummler" hin, der jedoch nahezu perfekt in die Weibull- und Lognormalverteilung passt.

2.2. Probability Plots interpretieren

Die Interpretation von Probability Plots hängt u.a. vom gefundenen Verteilungstyp ab. Die häufigsten Verteilungen in Usability Tests sind Lognormal-, Exponential- und Weibull-Verteilung, in dieser Reihenfolge. Welche Verteilung vorliegt, kann man an den entsprechenden Verteilungs-spezifischen Plots prüfen: sind die Datenpunkte hinreichend linear angeordnet, kommt die betreffende Verteilung infrage. Um das genauer zu prüfen, können wir im Plot jeweils eine Regressionsgerade hinzufügen. Der zugehörige R^2-Wert (d.h. der Anteil der durch das Regressionsmodell erklärten Varianz) ermöglicht einen einfachen Signifikanztest: NIST/SEMATECH (2012d) gibt eine Tabelle mit kritischen Werten hierzu an. Ist das beobachtete R^2 kleiner als der kritische Wert, ist die Hypothese zu verwerfen, dass die beobachtete Verteilung der angenommenen entspräche.

Im nächsten Schritt kann der Plot auf Ausreisser inspiziert werden. Als Ausreisser kommen Datenpunkte infrage, die nicht in die Verteilung der übrigen Datenpunkte passen. Im Plot ist das unmittelbar sichtbar: die kritischen Punkte liegen deutlich abseits der Regressionsgeraden, und zwar deutlich mehr als die übrigen Punkte, die zufällig um die Regressionsgerade verstreut sind. Oft ist das bei besonders schnellen oder langsamen Benutzern der Fall – potenziellen „Schummlern" oder „Bummlern", bei denen man seine Aufzeichnungen genauer auf weitere Auffälligkeiten inspizieren sollte.

Zur weiteren Interpretation und Parameterschätzung werden Ausreisser entfernt und die Plots neu berechnet.

Hat man ein passendes Verteilungsmodell gefunden, kann der entsprechende Plot weiter ausgewertet werden. Die Regressionsgerade bietet ja ein einfaches lineares Modell der gesamten Stichprobenverteilung! Da wir Zeiten und Prozentränge gegeneinander geplottet haben, können wir direkt ablesen, welcher Prozentsatz von Benutzern zu einer gegebenen Zeit die Aufgabe gelöst hat, bzw. welche Erfolgsquote wir zu einer bestimmten Zeit erwarten können. Weil dazu allerdings die Skalentransformationen der Achsen zurückgerechnet werden müssen, empfiehlt es sich, interessante R-Werte bzw. Perzentile (z.b. 0.05, 0.5 und 0.95, also 5, 50 und 95% der Benutzer) von vornherein in den Plot einzuzeichnen; die Schnittpunkte mit der Regressionsgeraden geben die entsprechenden Zeiten an.

Die weitere Interpretation und Parameterschätzung hängt von der gefundenen Verteilung ab.

2.2.1. Lognormalverteilung

Findet man eine Lognormalverteilung, erhält man eine Normalverteilung, indem man seine Daten einfach logarithmiert. Mit diesen logarithmierten Bearbeitungszeiten kann man dann ohne weiteres parametrische statistische Tests wie t-Test und Varianzanalyse ausführen und auf Signifikanz prüfen, muss allerdings zur Beurteilung von Unterschieden auf die ursprüngliche lineare Skala zurückrechnen. In diesem Fall ist es auch völlig gerechtfertigt, wie von Sauro (2011) empfohlen das geometrische Mittel als Statistik zu verwenden, da es gleich dem arithmetischen Mittel der Logarithmen, zurücktransformiert auf die lineare Skala ist. Im Plot entspricht diese Zeit dem Schnittpunkt der Regressionsgeraden mit der Zeitachse, allerdings mit einem wichtigen Unterschied: durch das censoring berücksichtigt der Plot auch die Information von nicht erfolgreichen Benutzern, die man bei rein rechnerischer Bestimmung verwerfen müsste. Nur bei 100% Erfolgsquote sind beide Werte gleich; bei geringeren Erfolgsquoten ergibt der Plot eine längere Zeitdauer.

2.2.2. Exponentialverteilung

Exponentialverteilungen sind typisch für Zeit-Intervalldaten. Sie sind charakteristisch für Prozesse wie z.B. radioaktiven Teilchenzerfall, Zeitintervalle zwischen Call Center-Anrufen etc., bei denen Ereignisse unabhängig voneinander und zufällig eintreten. Mathematisch lässt sich die Exponentialverteilung vollständig durch einen einzigen Parameter λ beschreiben. λ wird auch als **Ausfallrate** bezeichnet, da es in der Zuverlässigkeitsanalyse den Anteil der Teile beschreibt, die in einem gegebenen Zeitintervall ausfallen – und dieser Anteil ist bei Exponentialverteilungen konstant.

Im Probability Plot von exponentialverteilten Usability-Testdaten (Beispiel: Abb. 2) findet man typischerweise eine Regressionsgerade, die die Zeitachse nicht im Ursprung schneidet, sondern zu einer Zeit

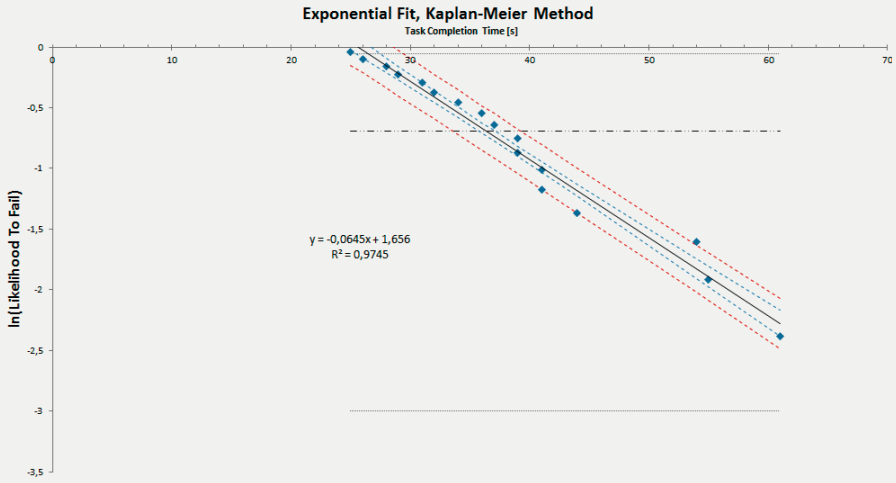

Abb. 2.
Probability Plot für exponentialverteilte Bearbeitungszeiten mit λ=0,0645 und t_0=25,7s. Die horizontalen Linien entsprechen dem 5., 50. und 95. Perzentil der erwarteten Verteilung. Die Konfidenzbänder der Regressionsgeraden stellen die Schätzungs- bzw. Vorhersagegenauigkeit dar. Das Exponentialmodell erklärt R^2=97,5% der beobachteten Varianz.

t_0. Die Exponentialverteilung ist daher nicht rein (und rechnerisch daher leicht zu übersehen), sondern um t_0 verschoben. Erst nachdem die Zeit t_0 verstrichen ist, beobachtet man eine konstante **Lösungsrate** λ, d.h. einen pro Zeiteinheit konstanten Anteil Benutzer, die die gestellte Aufgabe lösen. λ entspricht auch mathematisch der Steigung der Regressionsgeraden: ist sie steil, löst pro Zeiteinheit ein größerer Anteil der Benutzer die Aufgabe; ist sie flach, entsprechend weniger.

Das Exponentialverteilungsmodell teilt die beobachteten Zeiten damit mathematisch in einen **konstanten** Anteil t_0 und einen **Zufallsprozess** mit der Lösungsrate λ ein. Wenn dieses Modell die Daten korrekt abbildet, was in der Regel der Fall ist, was bedeuten dann t_0 und λ in der Realität?

t_0 liegt typischerweise nahe an der minimalen Bearbeitungszeit, d.h. der Zeit der schnellsten Benutzer, die nicht geschummelt haben, bzw. der Zeit, die man braucht, um als Experte die Aufgabe auf dem Idealpfad einfach durchzuklicken. Es ist plausibel, dass sich diese Zeit nicht weiter minimieren lässt, sondern durch physikalische Grenzen wie Systemreaktionszeiten und motorische Abläufe bestimmt wird, also die **mechanische Effizienz** der Benutzungsoberfläche beschreibt. [Abb. 2]

Der zu t_0 aufsetzende Zufallsprozess umfasst all die Zeitanteile, die zufälligen Schwankungen unterworfen sind. Zwar ist das auch bei motorischen Abläufen der Fall, doch ist dieser Varianzanteil vernachlässigbar gegenüber der Zeit, die Benutzer damit verbringen, Funktionen zu suchen, Fehler zu machen und zu korrigieren, und überhaupt die Benutzungsoberfläche zu verstehen. Dieser letztere Zeitanteil ist hoch variabel und entscheidend geprägt durch die Verständlichkeit der Benutzungsoberfläche. Die Lösungsrate λ ist damit ein direktes Maß für die **kognitive Effizienz** der Benutzungsoberfläche.

Abbildung 3 zeigt schematisch überlagerte Probability Plots von drei Apps. Die Apps A und B weisen denselben Achsenschnittpunkt t_0 auf, doch hat A eine deutlich höhere Lösungsrate λ. Würde man nur die Mittelwerte betrachten, würde A zwar besser abschneiden, der Unterschied würde jedoch nicht statistisch signifikant, da sich die beiden Verteilungen stark überlappen. Tatsächlich führt die geringere Lösungsrate von B zu einer größeren Streuung

und damit Standardabweichung der Zeiten. Anders ausgedrückt: je schlechter die Anwendung ist, desto schwerer ist es, diese Tatsache statistisch signifikant nachzuweisen!

Betrachten wir die Apps B und C: hier würde man überhaupt keinen Unterschied der Mittelwerte feststellen, beide sind gleich. B hat jedoch eine deutlich geringe Lösungsrate, während C längere Klickpfade und/oder schlechtere Systemperformance aufweist.

Die Apps B und C zeigen ein typisches Dilemma im UI Design: sind aufgeräumte, einfache Screens längere Klickpfade wert? Sollte man eher in die Performance von C oder in das Interaktionsdesign von B investieren? Da beide Faktoren hier getrennt visualisiert sind, kann das Designteam informierte Entscheidungen treffen. Performance-Verbesserungen von B, wie sie ein technologiefixiertes Team möglicherweise vorschlagen würde, können leicht als ungeeignet erkannt werden: sie wären teuer (B ist schon performant) und uneffektiv (weiterhin würden sehr lange Bearbeitungszeiten vorkommen, wie man in der unteren Hälfte des Plots sieht). **[Abb. 3]**

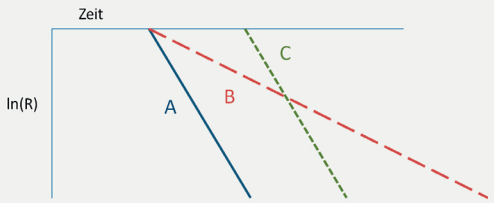

Abb. 3.
Schematisch überlagerte Probability Plots (Exponentialverteilung) für drei Apps.

2.2.3. Weibull-Verteilung

Während Exponentialverteilungen eine konstante Lösungsrate aufweisen, ist die Lösungsrate bei Weibull-Verteilungen einer systematischen Zu- oder Abnahme unterworfen. Typischerweise zeigen Weibull-verteilte Bearbeitungszeiten eine stetige Abnahme der Lösungsrate, was sich im Exponentialplot als Abflachung der Datenkurve und im Weibull-Plot als gerade Linie zeigt. Eine mögliche Ursache sind bei überlangen Aufgaben zunehmende Ermüdung, Frustration und Konzentrationsverlust.

Der Weibull-Plot ist besonders informativ, wenn überlange Bearbeitungszeiten auftreten und es nicht sicher ist, ob man es bei besonders langsamen Benutzern mit Ausreißern („Bummlern") zu tun hat. Passen die Zeiten in ein Weibull-Modell, d.h. der Plot ist linear und auch die „Bummler" liegen auf der Regressionsgeraden, deutet das auf eine **systematische** Abnahme der Lösungsrate hin, der **alle** Benutzer unterworfen sind. Ausreißer wären ein test-methodisches Problem; eine Weibull-Verteilung deutet vielmehr auf ein grundsätzlicheres Problem der getesteten Anwendung hin.

2.2.4. Verteilungsformen im Vergleich

Erste interne Analysen deuten darauf hin, dass Lognormal- und Exponentialverteilungen in deutlich über 80% der Fälle anzutreffen sind; oft lassen sich beide Modelle innerhalb der Signifikanzgrenzen anpassen. Weibull-Verteilungen sind mit etwa 15% deutlich seltener. Dass es hier nicht nur um reine Statistik geht, wird deutlich, wenn man überlegt, welche Mechanismen denn zu der einen oder anderen Verteilung führen können.

Die Exponentialverteilung setzt am wenigsten Annahmen voraus: man würde sie bei einem reinen Zufallsprozess erwarten, bei dem jeder Benutzer sozusagen eine Zufallsstichprobe an Usability-Problemen zieht, die mehr oder weniger viel Zeit kosten. Die Lösungsrate λ beschriebe die relative Häufigkeit und zeitlichen Kosten dieser Usability-Probleme, wie sie in der Gesamtheit der Aufgabe bei gegebener Oberfläche auftreten. Auch die Verschiebung um t_0 wäre durch die technischen Gegebenheiten (Systemantwortzeit + „Durchklick"-Zeit) hinreichend erklärt.

Bei einer Weibull-Verteilung käme ein stetig zunehmender, negativer Einfluss auf die Bearbeitungszeit hinzu. In der Zuverlässigkeitsanalyse sind Weibull-Verteilungen ein Indikator für Materialermüdung, die zu einer stetigen Zunahme von Fehlern führt. In unserem Fall hätte Benutzerermüdung den mathematisch umgekehrten, aber konzeptionell völlig analogen Effekt, nämlich die stetige Abnahme der Lösungsrate.

Bei der Lognormalverteilung sind die Verhältnisse weniger klar. In der Zuverlässigkeitsanalyse findet man Lognormalverteilungen bei Prozessen, in denen Fehlerquellen sich multiplikativ verhalten, sich also gegenseitig voraussetzen. Derartige Abhängigkeiten sind auch bei Teilaufgaben in einem Usability Test vorstellbar, aber schwierig zu analysieren.

Für die Usability-Praxis hat das Exponentialverteilungsmodell gegenüber dem Lognormalverteilungs-modell entscheidende Vorteile. Die Modellparameter t_0 und λ haben plausible Entsprechungen in der Realität und erlauben eine sehr einfache Berechnung von Erfolgsquoten bei gegebener Bearbeitungszeit bzw. umgekehrt den Zeiten, die man für eine gegebene Erfolgsquote veranschlagen muss. Zwar lassen sich auch im Lognormalverteilungsmodell statistische Tests und Parameter relativ leicht berechnen, doch kann die logarithmische Skala zu Fehlschlüssen führen. So entsprechen Differenzen in der logarithmischen Skala Proportionen in der linearen Skala – die Bedeutung einer Standardabweichung hängt also davon ab, wo auf der Skala sie abgetragen wird. Das Gleiche gilt für Unterschiede in der Bearbeitungszeit, die man z.B. bei verschiedenen Designalternativen misst.

Besonders problematisch ist die Beurteilung langer Bearbeitungszeiten, der „Bummler". Auch bei Lognormalverteilungen sind solche Daten statistisch zu erwarten; sie können leicht um ein Vielfaches über der mittleren Bearbeitungszeit liegen. Verwendet man – statistisch korrekt – das geometrische Mittel, das in Lognormalverteilungen noch kleiner als das arithmetische Mittel ist, sehen solche

Bearbeitungszeiten für den Laien wie grotesk schlechte Leistungen aus, obwohl sie statistisch alles andere als auffällig sind. Da jede Zeitmessung auch eine Leistungsmessung beinhaltet, stellen sich hier gerade im Umfeld von Geschäftssoftware offensichtliche ethische Anforderungen an die korrekte Kommunikation von Testdaten.

3. Ein typischer Analyseablauf

Probability Plots lassen sich mit Tabellenkalkulations-Software mit akzeptablem Aufwand soweit vorbereiten, dass der Analyseablauf weitgehend automatisiert wird. Eine xls-Datei kann beim Autor angefordert werden. Als Eingangsgrößen braucht man, für eine gegebene Aufgabe, für jeden Benutzer die Bearbeitungszeit der Aufgabe sowie einen binären Indikator, ob die Aufgabe erfolgreich gelöst wurde oder nicht. Diese Daten werden nach der Bearbeitungszeit aufsteigend sortiert und in separate Spalten der Tabelle einkopiert; Plots werden dann automatisch erzeugt.

Empfohlen werden zwei separate Arbeitsblätter: ein Übersichtsblatt mit Plots für Exponential-, Lognormal- und Weibull-Verteilung zur Identifikation der Verteilung, sowie ein weiteres mit einem detaillierten Exponential-Plot zur Parameterschätzung, der u.a. die Perzentile 5, 50 und 95 sowie Konfidenzgrenzen für die Regressionsgerade anzeigt.

Die Analyse beginnt mit dem detaillierten Exponential-Plot. Zunächst werden Ausreißer, Bummler oder Schummler identifiziert und ggfs. entfernt. Gibt es Bummler, prüfen wir auf dem Übersichtsblatt auf Weibull-Verteilung – passen die Bummler dort in die Verteilung, sind sie keine Ausreißer. Bei allen verdächtigen Datenpunkten werden die Aufzeichnungen genauer nach Auffälligkeiten durchsucht. Passt das Exponentialmodell, können in dem detaillierten Arbeitsblatt t_0 und λ abgelesen werden, und es geht weiter mit der nächsten Aufgabe.

Passt das Exponentialmodell weniger gut, können wir auf dem Übersichtsblatt schauen, welche Verteilung am besten passt – anhand der Krümmung der Datenpunkte sowie den R^2-Werten der Regressionsgeraden in den jeweiligen Plots. Diese Verteilungsanalyse erlaubt uns, testbare Hypothesen über Ursachenmechanismen aufzustellen, sowie angemessene Parameter und weitere Analyseverfahren zu identifizieren.

Für vergleichende Analysen werden Plots zunächst separat erstellt und anschließend überlagert; bei Exponentialplots können t_0 und λ so direkt verglichen werden. Wichtig ist hier auch die Prüfung auf Lognormalverteilung: kann ein Lognormal-Modell angepasst werden, sind parametrische Tests mit logarithmierten Daten statistisch korrekt durchführbar.

Wie alle statistischen Verfahren beschreiben Probability Plots die beobachtete Stichprobe, nicht die Grundgesamtheit. Alle Parameterschätzungen werden damit umso genauer, je größer die Stichprobe an Benutzerdaten ist. Probability Plots werden damit den größten Nutzen bei unmoderierten Online-Usability Tests sowie automatisch erfassten Verhaltensdaten entfalten. Da es dort auch auf effiziente Identifikation problematischer Daten ankommt, bietet die Datenvisualisierung mit Probability Plotting eine wirkungsvolle Unterstützung.

Literatur

1. ISO (1998). Ergonomic requirements for office work with visual display terminals (VDTs) Part 11: Guidance on Usability. ISO 9241–11:1998 (E)
2. ISO (2006). Software Engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Common Industry Format (CIF) for usability test reports. ISO/IEC 25062:2006(E)
3. Sauro, J. & Lewis, J.R. (2009). Correlations among Prototypical Usability Metrics: Evidence for the Construct of Usability. Proc. CHI 2009, ACM Press
4. Sauro, J. (2011): 10 Things to Know about Task Times. Retrieved May 2013 from http://www.measuringusability.com/blog/task-times.php
5. Luce, R.D. (1986). Response times: their role in inferring elementary mental organization. Oxford psychology series. No.8
6. Sauro, J. & Kindlund, E. (2005): How Long Should a Task Take? Identifying Specification Limits for Task Times in Usability Tests. In Proceeding of the Human Computer Interaction International Conference (HCII 2005), Las Vegas, USA
7. NIST/SEMATECH (2012a). e-Handbook of Statistical Methods. Retrieved May 2013 from http://www.itl.nist.gov/div898/handbook/. National Institute of Standards and Technology
8. NIST/SEMATECH (2012b). Probability Plotting. In: e-Handbook of Statistical Methods. Retrieved May 2013 from http://www.itl.nist.gov/div898/handbook/apr/section2/apr221.htm. National Institute of Standards and Technology
9. NIST/SEMATECH (2012c). Empirical model fitting – distribution free (Kaplan-Meier) approach. In e-Handbook of Statistical Methods. Retrieved May 2013 from http://www.itl.nist.gov/div898/handbook/apr/section2/apr215.htm#Modified K – M. National Institute of Standards and Technology
10. NIST/SEMATECH (2012 d) Critical Values of the Normal PPCC Distribution. In e-Handbook of Statistical Methods. Retrieved December 2012 from http://www.itl.nist.gov/div898/handbook/eda/section3/eda3676.htm. National Institute of Standards and Technology

[1] Da diese Linearität das entscheidende Kriterium ist, können die vertikale und horizontale Achsenzuordnung zur leichteren Interpretierbarkeit frei so gewählt werden, dass bestimmte Parameter einfacher abzulesen sind; z.B. kann statt R(t) auch das Komplement 1-R(t) geplottet werden.