NCP no common people

Smarta Stickprov

(Smart Sampling)

Statistisk metodik för kvalitetssäkring

Fredrik Carlemalm

fredrik.carlemalm@ncpab.se

Inledning

NCP no common people

Tester tar traditionellt mycket tid



- Men vi har inte tillräckligt med tid/pengar/folk!

Testarbetet går fortare med statistiska metoder

- Stickprovsteknik är extremt kraftfull och tjänar mycket tid

Testresultaten blir objektiva och kvantifierbara, dessutom!

Vi kan testa mindre och ändå få reda på mer än vi brukar

- Och vi kan leverera resultat mycket snabbare…

Tester tar traditionellt mycket tid

I utvecklingsprojekt - och än mer i förvaltning av IT-system - är det vanligen ont om kalendertid, pengar och/eller resurser.

Detta har blivit än värre i dagens "slimmade" organisationer, där det ofta är svårt att få testare från en stressad verksamhet. Och tester tar i sin traditionella form mycket tid.

Testarbetet går fortare med statistiska metoder

Trots den allmänna uppfattningen att man inte kan snabba på testarbetet finns det alternativ till fullskaliga, traditionella tester.

Ett sätt är med anpassad stickprovsteknik, s.k. statistisk testning, som innebär att vi vet tillräckligt väl, att det är tillräckligt rätt.

Testresultaten blir objektiva och kvantifierbara

Än bättre: vi får ett objektivt, mätbart kvalitetsmått i testerna! Vi kan alltså testa mindre – och veta betydligt mer.

- **Vi kan lättare – och snabbare – svara på om ett system tycks fungera.**

"It is not enough to do your best;
you must know what to do, and then do your best."

— William Edwards Deming

Vad innebär stickprovsmetodik?

NCP no common people

Ett slumpvis urval från en population

- Enheterna bör vara jämförbara vad gäller storlek, typ, etc.

Ett antal enheter i stickprovet som matchar ambitionsnivån

- För 99 % kvalitetsnivå (max 1 % fel) kan det räcka med 32 (!) i urvalet
- ✓ Populationens storlek påverkar inte stickprovets storlek (så mycket)!

Resultatet innebär att hela populationen accepteras eller förkastas!

Ett slumpvis urval från en population (ett antal enheter)

Enheterna bör vara jämförbara vad gäller storlek, typ, etc.
(t.ex. krav eller testfall med en viss prioritet, kodändringar, osv.).

Ett antal enheter i stickprovet som matchar ambitionsnivån

Vill vi vara 95 % säkra på att det är max 1 % fel, kan 32 - 50 st i urvalet räcka.
Siktar vi å andra sidan på 99,9 % kvalitet, behöver vi betydligt fler (men inte 500).

Populationens storlek påverkar inte stickprovets storlek (i nämnvärd omfattning).
Ett bestånd på en miljard enheter kräver inte ett större stickprov än för en miljon!

Resultatet innebär att hela populationen accepteras eller förkastas!

Testerna är alltså med stickprovsteknik bara kontrollerande, aldrig ”avlusande”.

När kan stickprov användas?

NCP no common people

För att ha nytta av stickprov förutsätts:

1. Relativt stora mängder
2. Likartade enheter
3. Verifierbarhet
4. Rimlig tillförlitlighet
5. Kontroll, inte avlusning

Inom krav- och testområdena stämmer det bra:

- Många krav att granska och verifiera
- Många testfall att ta fram och granska
- Många testfall att köra och verifiera

För att man ska kunna använda och ha nytta av stickprov förutsätts att:

1. Relativt stora mängder av något ska undersökas
2. Det som ska undersökas är av någorlunda likartad natur
3. Om en undersökt enhet är felaktig ska gå att avgöra
4. Önskad tillförlitlighet ska vara rimlig, inte total säkerhet!
5. Undersökningen ska vara av kontrollerande typ (och inte rättande)

Det betyder att stickprov kan vara till stor nytta särskilt på testområdet, där det ofta finns stora mängder testfall att köra och verifiera mot facit.

Detsamma gäller inom kravområdet, där krav ska granskas och senare verifieras mot den levererade lösningen, då inte bara i form av tester.

Skapade testfall behöver också granskas visavi bland annat kraven.

När kan stickprov användas? (forts.)

NCP no common people

Stickprov kan t.ex. också användas för:

- Granskning av process- och projektdokumentation
- Kontroll av års- och saldobesked
- Verifiering av aktiekurser

✓ När man har många – något hundratal – av någonting…

Stickprov används ofta IRL, men inte så ordnat:

- Alla kombinationer av data i inmatningsfält kan inte testas
- Alla varianter av flödesalternativ täcks sällan in

✓ **Med ordnade stickprov får vi kvantitativa resultat!!**

Stickprov kan användas närhelst man har en uppfattning om proportioner i ett stort antal – minst något hundratal – av någonting…

Stickprov används i praktiken ofta redan idag, om än inte så ordnat; i system- och acceptanstest går det inte att testa alla varianter av allt.

Med hjälp av t.ex. testmatriser säkerställer man då att man testar på bra sätt. Men med ordnade stickprov får vi en kvantifierbarhet.

Det kan t.o.m. vara ett alternativ till testautomatisering i t.ex. agil utveckling!

Vad tjänar vi på det?

NCP no common people

Typisk testanvändning:

- I teorin, i bästa fall: en faktor 5.000 eller mer!
- I ett normalfall: upp till faktor 10, alltid minst faktor 3

Styrande faktorer:

- Vilken kvalitetsnivå vi vill uppnå
- Hur säkra vi vill vara på att ha nått den
- Som vanligt: hur mycket vi kan och vill planera i förväg

Så varför gör vi inte alltid så här?

Typisk testanvändning:

I teorin, i ett extremfall, med ett stickprov på 200 bland 1.000.000 saker som ska verifieras, där man inte finner något fel: en faktor 5.000, så verifieringsarbetet blir försumbart!

I ett normalfall, med delsystem, fördjupade lokala tester inom vissa områden, etc. – upp till faktor 10, alltid minst faktor 3, enligt egen erfarenhet.

Styrande faktorer:

Vilken kvalitetsnivå vi vill uppnå
Kan variera mellan t.ex. olika delar av ett system

Hur säkra vi vill vara på att ha nått den nivån
Kan också variera mellan t.ex. olika delar av ett system

Som vanligt: hur mycket vi kan och vill planera i förväg
Men vi kan också använda metodiken för att kolla senare, vad man lyckats uppnå…

Så varför arbetar vi inte alltid så här?
Återkommer till detta senare – men tänk på vad som förväntas av t.ex. testare…

Formulera om frågorna!

NCP no common people

Typiska frågeställningar:

1. Hur många av kraven behöver jag kolla?
2. När vi nu kört *x* testfall utan allvarliga fel, räcker det?

Uttryckt som stickprov:

1. Hur stort stickprov bland de uppställda kraven behöver tas och kontrolleras för att statistiskt säkerställa den önskade kvalitetsnivån?
2. Givet ett visst stickprov och resultatet av detta, vilken kvalitetsnivå har nåtts, statistiskt sett?

(Ja, det LÅTER tråkigt…)

Teori: Skattningsmetoder (1)

NCP no common people

Ointresserad av statistik? Luta dig tillbaka och vila en stund!

Plocka ut n enheter bland totalt N. I det urvalet visar sig x vara felaktiga.
Hur stor är då totalt andelen felaktiga?

Punktestimat

$p \approx x / n$ *(Enkelt och intuitivt, ofta nog för en första skattning)*

Intervallskattning

Antag fortfarande $p \approx x / n$, men lite flexibelt:

$x / n - d_1 < p < x / n + d_2$, som täcker p med viss säkerhet.

(Det här är den professionellt bästa metoden – och med bäst stöd online…)

Ur ett bestånd om N likadana enheter – t.ex. samma slags programkodsändringar – tas ett stickprov som omfattar n enheter, där x av enheterna visar sig vara felaktiga. Hur stor är totalt andelen felaktiga? Dvs hur stor lär den okända felkvoten p vara?

Punktestimat

För tillräckligt stora stickprov kan man givetvis approximera p med x / n (= p*).
Men hur bra är det? Vågar vi lita på den skattningen?

Intervallskattning

Ett statistiskt sätt att kunna uttala sig om hur säkra vi är på vår approximation, fortfarande $p \approx x / n$, är att inrymma p i ett konfidensintervall:

$x / n - d_1 < p < x / n + d_2$, som med viss säkerhet täcker p.

Beroende på hur noga det är, väljer vi d_1 och d_2 så att vi täckt in p med t.ex. 95 eller 99 % säkerhet. Ju större säkerhet som krävs, desto större intervall behöver vi ta till.

För att vi ska kunna välja riktiga värden på d_1 och d_2 måste vi emellertid också veta något om hur sannolikheterna är fördelade i intervallet. Kommer det att vara lika stor chans att hitta x felaktiga enheter som att finna $x + nd_2$ eller noll stycken?

Teori: Skattningsmetoder (2)

NCP no common people

Hypotesprövning

Ett mer avancerat sätt att tänka är med hjälp av s.k. **hypotesprövning**.

Sätt upp hypotesen att $p <= p_0$, där p_0 är ett bestämt tal, 0 – 100 %, t.ex. 1 %.
Vi vill pröva hypotesen genom att använda stickprovet, så att vi accepterar hypotesen om $x <= x_0$, där x_0 är ännu ett bestämt tal, 0 – 100 %, t.ex. 0,5 %.

Detta förfarande kallas signifikanstest och är i vissa fall ett utmärkt alternativ till intervallskattning. För våra behov kommer dock intervallskattning att räcka bra (och är mycket enklare att använda). Dessutom finns det mycket bättre stöd online för intervallskattning.

Vilka matematiska formler som gäller och hur de kan approximeras beskrivs i appendix längst bak i bildmaterialet…

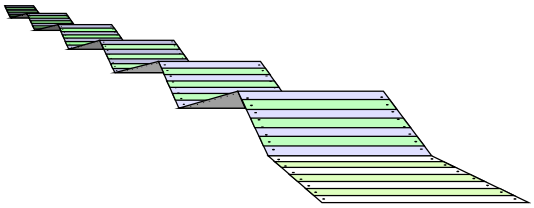
"Learning is not compulsory... neither is survival."
— William Edwards Deming

Not: Det är inte nödvändigt att p_0 och x_0 är lika, även om de i det enklaste fallet är det.

Praktik (förr): Använd en standard!

Med en lämplig standard kan mycket beräkningsarbete undvikas:

- ISO 2859
- ANSI / ASQ Z1.4
- BS6001
- DIN40.080
- NFX06-022
- UN148-42
- KS A 3109
- MIL-STD-105E



https://archive.org/details/MIL-STD-105E_1

Not: Vid nolltolerans, använd istället MIL-STD-1916.

Standard i stället för beräkning

Med en lämplig standard för stickprovstagning kan vi undvika egna kalkyler:

ISO 2859
ANSI / ASQ Z1.4
BS6001
DIN40.080
NFX06-022
UN148-42
KS A 3109
MIL-STD-105E ("pappa" och motsvarighet till alla de andra - men gratis!!)

MIL-STD-105E har i flera decennier använts av USA:s militär, men också civilt, för att kontrollera allt ifrån gevärspatroner till datamängder.

Formellt har 105E upphört att gälla och ersatts av Z 1.4 (som är motsvarande). Dock är det 105E, som man finner tillämpningar av online.

MIL-STD-1916 är en vidareutveckling, som dessutom är formellt gällande. I fokus för den är de tillämpningar som inte tillåter några fel.

Läsanvisning / -tips:

I synnerhet avsnitt 4.9.1 i standard MIL-STD-105E är av central praktisk betydelse. Avsnitten 3.1 - 3.3, 4.3 - 4.8, 4.10 och 4.12 bör också studeras.

Praktik (nu): Online-standard!

NCP no common people

Länk till automatiskt verktyg:
http://www.sqconline.com/military-standard-105e-tables-sampling-attributes

1. Bestäm storleken på beståndet (och dess ev. delar)
2. Bestäm acceptabel kvalitetsnivå, AQL
3. Bestäm inspektionsnivå, t.ex. II eller S-4
4. Bestäm typ av stickprov – 'single', 'double' eller 'multiple'
5. Skriv in dessa data, kör och läs av resultatet!

Not: SQC Online har nu även kalkylatorer för MIL-STD-1916 (i beta-version)

"In God we trust; all others (must) bring data."
"Without data, you're just another person with an opinion."

— William Edwards Deming

1. Bestäm storleken på beståndet och på dess eventuella delar
   - T.ex. antalet krav, funktioner, ändringar eller testfall

2. Bestäm acceptabel kvalitetsnivå, AQL.
   - AQL = 1 % motsvarar 95 % konfidens att beståndet är minst 99 % OK. Om felnivån är precis 1 % accepteras beståndet då 95 % av gångerna.

     Det finns även en annan nivå, LTPD (Lot Tolerance Percent Defective), som anger hur hög felnivån ska vara för att beståndet ska förkastas minst 90 % av gångerna. Praktiskt är det AQL som brukar anges för stickprovsplaner online, men AQL och LTPD hänger egentligen ihop genom planens s.k. OC Curve, Operating Characteristics. (Överkurs!)

3. Bestäm inspektionsnivå, i normalfall II (enligt standarden).
   - S.k. specialinspektioner kan snabba upp förfarandet. De minskar mängden av stickprov, men ökar även risken.

4. Bestäm typ av stickprov (single, double eller multiple; snart även sequential).
   - Med annat än 'single' får vi en bild redan vid första omgången.

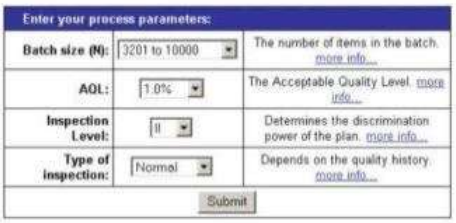
5. Läs av värden i automatkalkyl eller tabell för aktuellt stickprov.
   - Beroende på upplägg och resultat, ändra vid behov stickprovstyp.

Praktik: Typfall

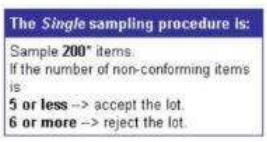
Normalinspektion, enhetsvis, singelprov

- Du har **50** delsystem med vardera ca **200** menyfunktioner implementerade.
  Du vill att de till minst **99 %** överensstämmer med kraven.
- Enligt Table I, General Inspection Level II, ska Sampling Plan L väljas.
  Enklaste sortens plan, Single Sampling Plan, anger ett stickprov på 200.
  Alla 10.000 funktionerna accepteras, om färre än sex fel hittas i stickprovet.

Eller mata in data i SQC Online …



.. och få ett resultat:



Med specialinspektion (nivå S-4), räcker det att ta ut 32 enheter! Men då får inga fel upptäckas.

Specialinspektion, enhetsvis, singelprov

Emellertid kan man även välja att göra en specialinspektion enligt någon av nivåerna S-1 – S-4 i tabellen, för s.k. specialinspektion.

(Detta förutsätter att man kan acceptera en större risk, åtminstone initialt.)

Nivån S-4 motsvarar plan G, där man med ett stickprov på 32 st når en nominell nivå på kvaliteten på 0,40 %.

Detta förutsätter dock att inga som helst fel tillåts för att enheten ska accepteras!

Tillämpningar IRL (1)

NCP no common people

Dokumentgranskning

- Du ska granska en kravspecifikation, **200** sidor, med i snitt **20** krav per sida. Projektledaren vill ha en snabb statusöversikt, med max **5 %** felaktigheter.

Förslag till lösning: Specialinspektion (S-4), dubbel inspektion, 20 i varje.
Godkännandenivå: Max 1 fel i inspektion 1, max 4 fel t.o.m. inspektion 2.

För att ge snabbt besked krävs alltså bara att 1 % av kraven kontrolleras!

Dokumentgranskning

Det här är att av de områden där jag haft mest nytta av smarta stickprov under åren.

Att granska planer, specifikationer, testfall, programkod etc. tar tid – fast det ofta är värt ansträngningen. Men åtminstone innan man går på djupet i dokumenten brukar det vara en mycket god idé att övergripande kontrollera kvaliteten.
Då är statistisk metodik till stor nytta.

Tillämpningar IRL (2)

NCP no common people

Kodändring – byte av valuta

- I de flesta systemen i en bank ska SEK i tillämpliga fall ersättas med EUR.
  Ändringarna görs i huvudsak i ett annat land, med halvmaskinellt kodstöd.
  Dessa levereras i omgångar om ca **20.000** ändringar vardera, totalt **30** ggr.
  Hur mycket ska kontrolleras för att säkra **99,9 %** kvalitetsnivå för allt detta?

  Förslag till lösning: Normalinspektion (II), enkel inspektion **per leverans**.
  Godkännandenivå: Max 1 fel i varje inspektion, som görs på 500 ändringar.

  Observera att det inte är lämpligt att verifiera alla leveranser samtidigt, framför allt av praktiska skäl – en leverans utgör här en typisk batch.
  Totalt blir det 30 x 500 = 15.000 ändringar som kollas – i bästa fall.

Verifiering av kodändringar (i STOR skala)

Valutaändringen har ju inte ägt rum i Sverige, men vi hade testplaner klara på en stor bank. En av ingredienserna skulle ha varit statistisk verifiering.

Något liknande skedde vid millennieskiftet.
Min konsultfirma (NCP) var en av få som vågade utfärda garantier till företag att de skulle ta sig in i 2000-talet utan problem. Detta berodde delvis på statistisk verifiering.

Tillämpningar IRL (3)

NCP no common people

Urval bland (täckande) testfall

- Det finns **1.750** menyfunktioner i ett system, som var en täcks i **6** testfall.
  Hur många fall behöver man färdigställa till testerna, köra och stämma av,
  för att uppnå en kvalitetsnivå på **99,8 %** - och i en inledande test minst **99 %**?

  Förslag till lösning, inledande test:
  Specialinspektion (S-4), dubbel inspektion, 32 i varje, för batchstorlek 10.500,
  med godkännandenivå: max 0 fel i inspektion 1, max 1 fel t.o.m. inspektion 2.

  Förslag till lösning, fullständig test:
  Normalinspektion (II), dubbel inspektion, 200 i varje, för batchstorlek 10.500,
  med samma godkännandenivå (!) (max 0 fel i inspektion 1, max 1 fel t.o.m. inspektion 2)

  Observera att det är klokt att ta fram och förbereda fler testfall än så.
  Om testfallen består av unikt täckande steg, kan färre fall behövas.
  Men i ett första skede 64 testfall, senare 400.

Urval bland (fullständigt) täckande testfall

Detta är ett exempel från telekom-världen – något modifierat för att skydda de oskyldiga.
I verkligheten var kvaliteten så usel att vi inte kom till fas 2, med den fullständiga testen.

Det var dock en av få gånger jag sett dokumenterat fullständigt täckande testfall.

Tillämpningar IRL (4)

NCP no common people

Redan genomförda tester – hur bra är kvaliteten?

- Man har kört ca **300** väl täckande testfall, om vardera **fem** unika teststeg. **Inget** enda allvarligt fel har påträffats under denna slutgiltiga testomgång. Vilken kvalitetsnivå kan man därigenom säga att systemet ifråga uppnått?

  Föreslagen lösning: Jämför med tabeller eller SQC online för 1.500 i batch. Godkännandenivå: 0 fel i inspektionen, vilket visar sig motsvara 99,99 %.

  För ett standardsystem en ganska överdriven kvalitetsnivå, kan man tycka. Jämför t.ex. med avtal på serverdrift, som ofta ger ca 99,8 % tillgänglighet.

  I systemutveckling åsätter man ofta en AQL på 1 % för allvarliga fel, medan mindre allvarliga fel får en AQL på 2,5 %.

Redan genomförda (täckande) tester

Ganska ofta får man som konsult inom test- och kvalitetsområdet komma in och hjälpa till när kunden redan har kört några mer eller mindre lyckade testomgångar. Då behöver man bedöma kvaliteten utifrån testrapporter m.m. för att kunna planera det fortsatta arbetet.

Det ger ju också gott rykte att få ut något av testresultaten som inte kunden har insett själv…

FAQ

- Beror inte stickprovet på mängden som ska kollas?
- Kan vi dela in det vi ska kolla i olika delar?
- Vad händer om vi hittar fel?
- Hur säkert är detta?
  Kan det verkligen stämma!?
- Så varför arbetar vi inte alltid så här?

"It is not necessary to change. Survival is not mandatory."

— William Edwards Deming

Beror inte stickprovet på systemstorleken?

Nej, inte nämnvärt. Med största tänkbara omfång räcker 50 enheter i stickprovet på nivå S4.

Vill vi ha högre säkerhetsnivå och kanske använda dubbla inspektioner, kan vi välja ca 100 enheter i stickprovet.

Om det skulle vara fråga om testfall, kan dessutom enheterna – testpunkterna – ofta kombineras till testfall, som det då kan behövas 20 – 30 av.

Kan vi dela in det vi ska kolla i olika delar?

Ja, det lär t.o.m. vara ett vanligt sätt att arbeta, bl.a. utifrån varierande kvalitetskrav.
Vi kan då även tidigt stickprova ett område, dokument eller delsystem som omfattar några procent av helheten och anpassa det fortsatta arbetet efter resultatet.

Vad händer om vi hittar fel?

Om vi hittar ett enda fel med enklaste formen av inspektion förkastar vi systemet, men använder vi en lite mer avancerad form av inspektion testar vi först djupare.

Om systemet inte håller måttet, måste det givetvis rättas och sedan testas om.

Hur säkert är detta? Kan det verkligen stämma!?

Både teorierna bakom och existerande standarder, bl.a. MIL-STD-105E, är stabila. Metodiken används också i många branscher, t.ex. i samband med hårdvaruimport.

Så varför arbetar vi inte alltid så här?

Tester används för ”avlusning” så sent som i systemtest, inte bara för kontroll.
Tekniken är inte känd inom mjukvaruutveckling (som inte är en så mogen bransch).

Ordnade granskningar förekommer inte ofta, så hur kan det bli snabbare än att inte göra något alls?
Hur kan något utmana "magkänslan" hos alla erfarna? (I bästa fall bekräftar det bara vad de tror.)

Det verkar också krångligt och osäkert om man inte känner den matematiska bakgrunden.

Och framför allt är det enklare att göra som man alltid gjort!

Det finns ingenting så svårt att ta itu med, ingenting så väldigt att leda, ingenting så osäkert i framgång, som att införa en ny tingens ordning, ty den som försöker har nämligen alla dem till fiender som drog fördel av den gamla ordningen och han har endast ljumma försvarare i dem som drar fördel av den nya. – N. Machiavelli

Nackdelar och faror

- Otydlig ambitionsnivå
- För hög eller låg ambitionsnivå
- Olika, ojämförliga enheter
- Missvisande stickprov
- Test/granskning kan inte användas för avlusning/upprättning
- Metodiken accepteras inte av berörda
- Felaktig tillämpning av stickprov

Ambitionsnivån (och mer specifikt AQL) måste vara tydlig i alla delar.
Annars definieras gärna de uppnådda resultaten som "good enough".
Dessutom får man inte "commitment" vare sig uppifrån eller nerifrån.

Nivån kan vara satt för högt eller lågt, av okunskap eller girighet.
Särskilt alltför glesa stickprov gör lätt att vi får en felaktig uppfattning…
En passande nivå beror på typen av system, dokument, område, etc. En pilotapplikation lär t.ex. inte kräva samma kvalitetsnivå som ett livsuppehållande system i sjukvården.

En god idé är ofta att jämföra med service-nivåerna från ett SLA – ofta förväntas servrar fungera med 99,8 % tillgänglighet. En AQL på 99,99 % är då för hög och en på 95 % för låg.

Ett system eller område kan bestå av flera olika, icke jämförbara enheter.
I så fall, dela upp det i mindre delar, eller se på det ur en annan synvinkel!

Vi kan ha otur – eller förstått fel – och ta ett missvisande stickprov.
T.ex. stickprov bland testfall, som totalt ej täcker systemfunktionaliteten...
Säkerställ att det du tar ditt stickprov ifrån är hela populationen –
eller i vart fall en sann, proportionell avbildning av denna!

Test/granskning kan inte användas för avlusning/upprättning
Många organisationer tror fortfarande att det är avsikten med sådana aktiviteter.
Det är ett viktigt skäl till varför stickprov inte accepteras. (Vadå "förkasta"!?)

Ändå kan man naturligtvis hjälpa till med att höja kvaliteten, när den befunnits för låg, genom att rätta upp dokument, nöta igenom kod etc. - men inte inom egen budget!
I en agil värld ska det förhoppningsvis inte vara något problem, hur som helst.

Metodiken accepteras inte av testarna, kravanalytikerna, projektledaren (eller andra intressenter)
Detta är troligen huvudanledningen till varför stickprov inte används så ofta. Metodiken är som sagt inte välkänd; det är därför jag är här. Ni lär behöva starta med några "lågt hängande frukter" (t.ex. genom att snabbt avgöra om en kravspecifikation håller måttet).

Otillräckliga kunskaper i statistik gör att stickprovstekniken tillämpas fel
Men om du kan läsa och förstå MIL-STD-105, är du på god väg (och det finns hjälp att få, t.ex. av mig!)

Appendix: Statistisk metodik (1)

Dragning utan återläggning

- Att ta stickprov bland ett antal (likartade) kodändringar för att se om något av stickproven är felaktigt motsvarar ett av statistikernas favoritexempel; att blint plocka ut olikfärgade kulor ur en urna, utan återläggning, för att efteråt se vilka kulor som har en viss av färgerna.
  Talar vi om två färger, t.ex. svart och vit, kan felaktigheter sägas motsvara svarta kulor.

Hypergeometrisk fördelning

- Det vi då har är en hypergeometrisk fördelning, med en sannolikhet $p_X(k)$ enligt följande formel:

$$p_X(k) = \binom{Np}{k}\binom{Nq}{n-k}/\binom{N}{n};\ 0 <= k <= Np,\ 0 <= n-k <= Nq$$

$$\binom{Np}{k} = \frac{(Np)!}{k!(Np-k)!},\ etc.$$

Appendix: Statistisk metodik (2)

Binomialapproximation

- För stora N - och förutsatt att kvoten n / N < ca 10% - kan ovanstående formel approximeras med en binomialfördelning; $p_X(k) = Bin(n, p)$:

$$p_X(k) \approx \binom{n}{k} p^k q^{n-k} \; ; \; 0 <= k <= n$$

Normalapproximation

- För stora n och N kan på motsvarande sätt en normalapproximation (dvs den så nyttiga normalfördelningskurvan) användas:

$$P(a < X <= b) \approx \Phi\left(\frac{b-np}{\sqrt{npq}}\right) - \Phi\left(\frac{a-np}{\sqrt{npq}}\right) \; ; \; X \in Norm(np, \sqrt{npq})$$

- Detta förutsätter att standardavvikelsen är tillräckligt stor, vilket i praktiken ställer krav på att npq > ca 10
- Vanligen förfinar man ytterligare approximationen med s.k. halvkorrektion

Appendix: Statistisk metodik (3)

Poissonfördelning

- Om p är litet och N mycket större än n, så att p + n / N < ca 10 %,
  kan vi approximera p_X (k) med en Poissonfördelning, Po (np):

$$p_X(k) \approx \frac{e^{-np}(np)^k}{k!} = Po\ (np);\ k = 0, 1, 2, \ldots$$

Poisson-/normalfördelning

- Är dessutom np tillräckligt stort, minst ca 15, med värdet på p fortfarande litet och därmed
  q = ca 1, kan vi i formlerna för Normalapproximation ersätta npq med np och sålunda få:

$$P\ (a < X <= b) \approx \Phi\left(\frac{b + ½ - np}{\sqrt{np}}\right) - \Phi\left(\frac{a + ½ - np}{\sqrt{np}}\right),\ med\ halvkorrektion;\ X \in Norm\ (np, \sqrt{np}\ )$$

- Detta är nog att betrakta som överkurs…
  Det viktigaste är vilka möjligheter vi har att använda oss av normalfördelningen.

Appendix: Normalfördelning

Normalapproximation

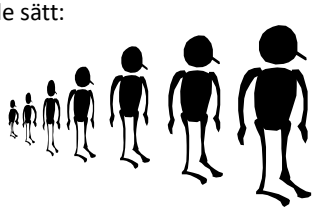
$$P(a < X <= b) \approx \Phi\left(\frac{b-np}{\sqrt{npq}}\right) - \Phi\left(\frac{a-np}{\sqrt{npq}}\right);\ X \in Norm(np, \sqrt{npq})$$

- I praktiken använder vi vanligen formeln med b = ∞ eller a = b, vilket kan slås upp i statistiktabeller och skrivs på följande sätt:

  $P(X > \lambda_\alpha) = \alpha$, *för* $b = \infty$

  respektive

  $P(-\lambda_{\alpha/2} < X < \lambda_{\alpha/2}) = 1 - \alpha$, *för* $a = b$



- Vanliga och för oss användbara värden på a är 5 %, 1 % och 0,1 % - typiska acceptabla felnivåer.

Appendix: Intervallskattning

Konfidensintervall, normalfördelning

- Om vi vill sätta upp ett konfidensintervall för den nyss nämnda fördelningen, får vi (med normalapproximation, $X \in Norm(np, \sqrt{npq}), X/n \in Norm(p, \sqrt{pq/n})$):

  $I_p = (p^* - \lambda_{\alpha/2}\sqrt{p^*(1-p^*)/n}\ ,\ p^* + \lambda_{\alpha/2}\sqrt{p^*(1-p^*)/n})$

- Motsvarande för en fördelning nära nollpunkten, med ett enkelsidigt intervall:

  $I_p = (0, p^* + \lambda_{\alpha}\sqrt{p^*(1-p^*)/n})$

Exempel

- Antag att vi skulle finna fem fel i ett stickprov på 250. Uttrycket under roten blir då 0,02 x 0,98 / 250, vilket ger ett värde på rotuttrycket på ca 0,00885.
- Skulle vi nu vilja uttrycka oss med 95 % säkerhet, letar vi i en normalfördelningstabell upp α = 0,05 (eller snarare $\alpha/2$ = 0,025) och får $\lambda_{\alpha/2}$ = 1,9600.
- Med 95% säkerhet ligger då felkvoten i intervallet 0,020 ± 0,017, dvs 0,3 - 3,7 %.

Appendix: Stickprovsstorlek

Bestämning av stickprovsstorlek

- Genom att istället föreskriva intervallets längd, liksom konfidensnivån, kan stickprovsstorleken styras.
- För att detta ska bli effektivt bör vi dock ha en uppfattning om storleken på p*.
  I annat fall bör vi anta värsta möjliga utfall, att p* = 1 - p* = 0.5,
  så att det under rotuttrycket för intervallet står 0,25 / n.

Exempel

- Antag att vi med 99 % säkerhet ska kunna säga att p* är korrekt, med 1 % marginal åt varje håll.
  Rimligen är inte mer än 1 % av beståndet felaktigt; snarare rör det sig om en tiondels procent…

❖ Intervallängden $= 2 \times 0{,}01 = 2\,\lambda_{\alpha/2}\sqrt{0{,}01(1-0{,}01)/n} = (\alpha/2 = 0{,}005)\ 2 \times 2{,}5758 \times \sqrt{0{,}0099/n}$

$n = 0{,}0099 \times (2{,}5758 / 0{,}01)^2 = 656$

- Egentligen bör vi räkna med ett enkelsidigt intervall så nära nollpunkten, vilket ger:

❖ Intervallängden $= 0{,}01 = \lambda_{\alpha}\sqrt{0{,}01(1-0{,}01)/n} = (\alpha = 0{,}01)\ 2{,}3263 \times \sqrt{0{,}0099/n}$

$n = 0{,}0099 \times (2{,}3263 / 0{,}01)^2 = 536$

Motsvarande för 95 % säkerhet, 5 % marginal och max 5 % fel i beståndet ger *n* = 51. *np* blir då mindre än 15, vilket gör approximationen teoretiskt osäker, men praktiskt fungerar *n* > 20.