STK-IN1050 - Statistikk for informatikere

Gjennomgang av prøveeksamen

Matematisk institutt

2026-05-18

Oppgave 1: Korte begrepsspørsmål

1a. En systemutvikler har 15 målinger av responstiden til en server og ønsker å rapportere et usikkerhetsintervall rundt medianen av disse målingene. En kollega foreslår: «bruk bare standardformelen for konfidensintervaller.» Du forklarer at standardformelen er laget for gjennomsnittet og ikke direkte gjelder for medianen, og foreslår i stedet å bruke bootstrap.

Med egne ord — beskriv (i) hva et bootstrap-utvalg er, og (ii) hvordan en samling av bootstrap-medianer kan brukes til å konstruere et usikkerhetsintervall.

Løsningsforslag

1a. Et bootstrap-utvalg er et nytt datasett på 15 observasjoner trukket med tilbakelegging fra de opprinnelige 15 målingene. Hvis trekker mange (f.eks. 1000) slike utvalg og beregner medianen for hvert utvalg, har vi en samling av bootstrap-medianer. Vi kan da konstruere et 95% usikkerhetsintervall ved å ta intervallet mellom 2,5-persentilen og 97,5-persentilen i fordelingen av bootstrap-medianene.

Oppgave 1: Korte begrepsspørsmål

1b. To studenter diskuterer et histogram laget fra 200 simulerte utvalg. Student 1 sier: «dette histogrammet er sannsynlighetsfordelingen til den underliggende stokastiske variabelen.» Student 2 sier: «nei — det er en empirisk tilnærming til sannsynlighetsfordelingen.» Hvem har rett, og hvilken begrepsmessig forskjell er det som påpekes?

Løsningsforslag

1b. Student 2 har rett. Histogrammet er laget fra et simulert utvalg og viser derfor en empirisk fordeling. Den empiriske fordelingen er en tilnærming til den teoretiske sannsynlighetsfordelingen basert på et endelig utvalg.

Oppgave 1: Korte begrepsspørsmål

1c. En student konkluderer: «siden p-verdien i testen min er 0,03, er det bare 3% sjanse for at nullhypotesen er sann.» Forklar kort hva som er galt med denne konklusjonen. Formuler med dine egne ord hva en *p*-verdi på 0,03 faktisk betyr.

Løsningsforslag

1c. *p*-verdien er ikke sannsynligheten for at nullhypotesen er sann gitt dataene. En *p*-verdi på $0,03$ betyr at hvis nullhypotesen faktisk er sann, så er sannsynligheten for å observere noe like eller mer ekstremt enn det vi så i dataene, lik 3%.

Oppgave 1: Korte begrepsspørsmål

1d. En forsker bruker lineær regresjon for å modellere sammenhengen mellom to variabler og skriver: «stigningstallet jeg fikk er 2,3, så *X* forårsaker at *Y* øker med 2,3 for hver enhet.» Uten å bruke avanserte metoder: gi én konkret grunn til at denne kausale påstanden kan være feil.

Løsningsforslag

1d. Det kan finnes en tredje variabel Z som påvirker både X og Y. Da kan det se ut som det er en sammenheng mellom X og Y, selv om det egentlig er Z som forårsaker endringene i begge. I så fall beskriver regresjonene bare en assosiasjon, ikke nødvendigvis at endring i X faktisk forårsaker en endring i Y.

Oppgave 2: Sannsynlighet via simulering

En partikkel starter i posisjon 0. For hvert tidssteg, uavhengig av alle andre steg:

- partikkelen beveger seg til høyre (+1 posisjon) med sannsynlighet 0,55, og
- partikkelen beveger seg til venstre (-1 posisjon) med sannsynlighet 0,45.

2a. Skriv en Python-funksjon med følgende signatur:

```
def simulate_walk(n_steps):
    ...
```

Funksjonen skal simulere én tilfeldig vandring med `n_steps` steg fra startposisjon 0 og returnere partikkelens sluttposisjon som et heltall. Bruk kun standardmodulen `random`.

Løsningsforslag

2a.

```
import random

def simulate_walk(n_steps):

  position = 0

  for _ in range(n_steps):
    position += 1 if random.random() < 0.55 else -1

  return position
```

Oppgave 2: Sannsynlighet via simulering

2b. Skriv en Python-funksjon med følgende signatur:

```
def prob_above_threshold(n_steps, threshold, n_sim):
    ...
```

med argumenter:

- `n_steps` (int): antall steg i hver simulert vandring,
- `threshold` (int): posisjonsgrensen å sjekke mot,
- `n_sim` (int): antall uavhengig simulerte vandringer,

som returnerer ett flytall: den estimerte sannsynligheten for at partikkelen ender opp strengt over `threshold` etter `n_steps` steg. Funksjonen skal kalle `simulate_walk` fra 2a.

Under funksjonen din, skriv én setning som forklarer hva divisjonen i returverdien din beregner.

Løsningsforslag

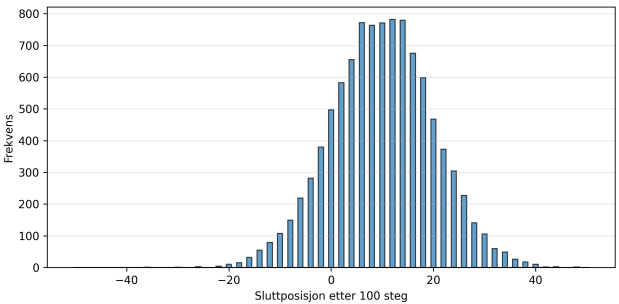
2b.

```
def prob_above_threshold(n_steps, threshold, n_sim):
    simulated_walks = [simulate_walk(n_steps) for _ in range(n_sim)]
    count = sum([1 for walk in simulated_walks if walk > threshold])
    return count / n_sim
```

Denne divisjonen beregner andelen av de `n_sim` simulerte vandringene som ender med sluttposisjon strengt over `threshold`, altså den estimerte sannsynligheten for å havne over grensen.

Oppgave 2: Sannsynlighet via simulering

2c. Figuren under viser et histogram over sluttposisjoner fra 10 000 simulerte vandringer med 100 steg hver.



(i) Sentrum av histogrammet er nær +10, ikke ved 0. Beregn for hånd den forventede sluttposisjonen til partikkelen etter 100 steg. Vis alle stegene i beregningen. (Hint: start med å beregne forventet forflytning per enkelt steg: $\mathbb{E}[\text{steg}] = 0,55 \times (+1) + 0,45 \times (-1)$.)

(ii) Selv om hvert enkelt steg er nøyaktig +1 eller -1, har histogrammet en klokkeform. Nevn det teoremet som forklarer hvorfor fordelingen av summen av mange uavhengige tilfeldige variabler har denne tendensen, og angi kort hva det sier.

Løsningsforslag

2c.

(i) La X være forflytningen i steg i. Da er

$$\mathbb{E}[X_i] = 0,55 \times (+1) + 0,45 \times (-1) = 0,55 - 0,45 = 0,1.$$

Sluttposisjonen er gitt ved $S_{100} = X_1 + X_2 + \cdots + X_{100}$. Vi har

$$\mathbb{E}[S_{100}] = \mathbb{E}[X_1 + X_2 + \cdots + X_{100}] = 100\,\mathbb{E}[X_1] = 100 \times 0,10 = 10.$$

(ii) Teoremet som forklarer klokkeformen er sentralgrenseteoremet. Teoremet sier: når man summerer et stort antall uavhengige, identisk fordelte stokastiske variabler med endelig varians, vil fordelingen til summen være tilnærmet normalfordelt, uansett den underliggende fordelingen til hver enkelt variabel.

Oppgave 3: Estimatorer og utvalgsfordeling

En fabrikk produserer treplanker som alle skal ha en sann lengde på 200 cm. Et måleinstrument registrerer lengden på hver planke, men legger til tilfeldig målefeil: hver måling er uavhengig lik sann lengde pluss en normalfordelt feil med forventning 0 og standardavvik 1,5 cm.

En kvalitetssjef måler én planke `n` ganger og bruker gjennomsnittet av disse `n` målingene som sitt estimat for plankens sanne lengde.

3a. Skriv en Python-funksjon med signaturen

```
def repeated_measure(true_length, n_measurements, sigma):
    ...
```

som simulerer `n_measurements` uavhengige målinger av en planke med gitt `true_length` og standardavvik til feilen `sigma`, og returnerer gjennomsnittet av disse som én flyttall-verdi. (Bruk `scipy.stats.norm.rvs` eller en tilsvarende metode.)

Løsningsforslag

3a.

```python
from scipy.stats import norm

def repeated_measure(true_length, n_measurements, sigma) -> float:

    def measurement_error(sigma) -> float:
        return norm.rvs(loc=0, scale=sigma, size=1)[0]

    measurements = [true_length + measurement_error(sigma) for _ in range(n_measurements)]
    return sum(measurements) / n_measurements

# alternatively:
def repeated_measure(true_length, n_measurements, sigma) -> float:
    measurements = norm.rvs(loc=true_length, scale=sigma, size=n_measurements)
    return measurements.mean()
```

Oppgave 3: Estimatorer og utvalgsfordeling

3b. Skriv en Python-funksjon

```
def simulate_estimates(true_length, n_measurements, sigma, n_repeats):
    ...
```

som returnerer en liste med `n_repeats` uavhengige lengdeestimater, der hvert estimat er produsert under de samme betingelsene som i 3a. Du kan anta at `repeated_measure` fra 3a allerede er definert og tilgjengelig.

Løsningsforslag

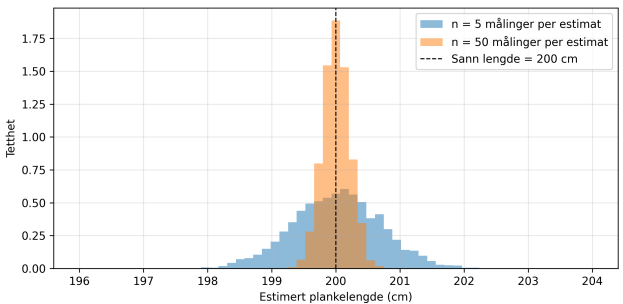
3b.

```
def simulate_estimates(true_length, n_measurements, sigma, n_repeats):
  return [repeated_measure(true_length, n_measurements, sigma) for _ in range(n_repeats)]
```

Oppgave 3: Estimatorer og utvalgsfordeling

3c. Figuren under viser to histogrammer produsert ved å kalle

```
simulate_estimates(200, 5,  1.5, 4000)   # blå
simulate_estimates(200, 50, 1.5, 4000)   # oransje
```



Diskuter kort hva denne figuren forteller oss om estimatoren «gjennomsnitt av n målinger» med hensyn på:

(i) forventningsskjevhet — overvurderer eller undervurderer estimatoren systematisk den sanne lengden 200?

(ii) varians / spredning — hvordan avhenger estimatorens spredning av n?

Løsningsforslag

3c. Vi vet at den sanne lengden er 200 cm, og hver måling er gitt ved $X_i = 200 + \varepsilon_i$ med $\mathbb{E}[\varepsilon_i] = 0$ og $\mathrm{SD}(\varepsilon_i) = 1,5$ cm. Estimatoren er gjennomsnitt av n målinger

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i.$$

(i) Vi har at

$$\mathbb{E}[\bar{X}] = \mathbb{E}\Big[\frac{1}{n}\sum_{i=1}^{n}(200 + \varepsilon_i)\Big] = 200 + \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\varepsilon_i] = 200.$$

Begge histogrammene ligger sentrert rundt 200. Det betyr at estimatoren er forventningsrett.

(ii) For gjennomsnittet gjelder at

$$\mathrm{Var}(\bar{X}) = \frac{1}{n^2}\sum_{i=1}^{n}\mathrm{Var}(X_i) = \frac{1}{n^2} \times n \times \sigma^2 = \frac{\sigma^2}{n},$$

så standardavviket er $\sigma/\sqrt{n}$.

Oppgave 3: Estimatorer og utvalgsfordeling

3d. Et nytt måleinstrument er forventningsskjevt: det legger til et systematisk avvik på +0,5 cm i tillegg til en normalfordelt tilfeldig feil med standardavvik 0,5 cm. Uten å skrive kode: beskriv med ord et histogram analogt til det oransje histogrammet i Figur 1, men basert på målinger fra dette nye instrumentet. Spesifikt: Hvor vil det nye histogrammet være sentrert, og hvordan vil bredden være sammenliknet med det oransje histogrammet? Du kan gjerne skissere for deg selv, men angi kun en kort verbal beskrivelse i svaret.

Løsningsforslag

3d. Det nye histogrammet vil ha samme form som det oransje histogrammet, men sentrum på fordelingen vil bli skjøvet opp med 0,5 cm. Histogrammet vil også bli smalere, dvs. ha mindre spredning, dersom feilen her har standardavvik 0,5 cm sammenlignet med 1,5 cm for the oransje histogrammet.

Oppgave 3: Estimatorer og utvalgsfordeling

3e. Hvorfor er det å ha lav varians ikke tilstrekkelig i seg selv for at en estimator skal være en god estimator? Henvis til svaret ditt i 3d for å gjøre poenget konkret.

Løsningsforslag

3e. Lav varians betyr at verdiene estimatoren tar ligger tett samlet, men sier ingenting om hvor de ligger i forhold til den sanne verdien. Resultatene i 3c(ii) viser at gjennomsnittet av mange målinger har lavere varians enn gjennomsnittet av få målinger, men hvis estimatoren er forventningsskjev som i 3d, da vil verdiene samle seg tett rundt en verdi som er forskjellig fra den sanne verdien. En god estimator trenger både lav varians og liten eller ingen skjevhet i forhold til den sanne verdien.

Oppgave 4: Hypotesetesting

Et firma kjører en test av to utforminger av en registreringsside. Hver besøkende til siden får tilfeldig tildelt én design og enten registrerer seg eller ikke. Etter testen observerer firmaet:

- Design A: 1500 besøkende, 102 registreringer
- Design B: 1500 besøkende, 138 registreringer

Teamet vil vurdere om sign-up-raten for design B reelt sett er høyere enn for A, eller om den observerte forskjellen på 36 registreringer kan forklares som tilfeldig variasjon under antakelsen om at de to designene er like gode.

4a. Formuler, med vanlige ord, en passende nullhypotese og en passende alternativ hypotese. Begrunn kort om en ensidig eller tosidig test er mest passende her, gitt spørsmålet til teamet.

Løsningsforslag

4a. Nullhypotese H_0: Den sanne registreringsraten er den samme for design A og design B. Eventuelle forskjeller i antall registreringer skyldes bare tilfeldigheter.

Alternativ hypotese H_a: Den sanne registreringsraten er høyere for design B enn for design A.

Ensidig test er mest naturlig her gitt det konkrete spørsmålet om sign-up-raten for design B reelt sett er høyere enn for A.

Oppgave 4: Hypotesetesting

4b. Skriv en Python-funksjon med følgende signatur:

```
def simulate_null_difference(n_visitors_per_group, p_null, n_sim):
    ...
```

som, under nullhypotesen om at begge design har en felles sann sign-up-rate `p_null`, simulerer `n_sim` versjoner av eksperimentet og returnerer en liste med `n_sim` heltall, der hvert heltall er `signups_B - signups_A` fra ett simulert eksperiment.

Argumenter:

- `n_visitors_per_group` (int): antall besøkende i hver gruppe, her 1500
- `p_null` (float): antatt felles sign-up-rate under H_0; i vårt tilfelle den pooled-estimerte $(102 + 138)/3000 = 0.08$
- `n_sim` (int): antall simulerte eksperimenter

Du kan bruke `scipy.stats.binom.rvs(n, p, size=1)` eller implementere med standardmodulen `random`.

Løsningsforslag

4b.

```
import random

def simulate_null_difference(n_visitors_per_group, p_null, n_sim):
    results = []
    for _ in range(n_sim):
        registrations_a = sum(random.random() < p_null for _ in range(n_visitors_per_group))
        registrations_b = sum(random.random() < p_null for _ in range(n_visitors_per_group))
        results.append(registrations_b - registrations_a)
    return results
```

Oppgave 4: Hypotesetesting

4c. Anta at `null_diffs` er en liste av lengde 100 000 returnert av `simulate_null_difference(1500, 0.08, 100000)`. Skriv noen få linjer Python-kode som beregner den ensidige p-verdien for observasjonen `observed_diff = 138 - 102 = 36` og lagrer den i en `float`-variabel `p`.

Løsningsforslag

4c.

```
observed_difference = 138 - 102
null_differences = simulate_null_difference(1500, 0.08, 100000)
p = sum(difference >= observed_difference for difference in null_differences) /
    len(null_differences)
```

Oppgave 4: Hypotesetesting

4d. Selv om p-verdien fra en slik test viser seg å være under 0,05, advarer teamets ledende ingeniør: «en liten p-verdi, alene, er ikke nok til å konkludere med at design B bør erstatte design A.» Forklar hva ingeniøren mener ved å nevne én antakelse bak hypotesetesten som bør vurderes før man tar beslutningen.

Løsningsforslag

4d. Hypotesetesten bygger på flere antakelser, og en liten *p*-verdi sier bare at dataene er lite forenlige med H_0 gitt at antakelsene holder. Én viktig antakelese som bør vurderes er at det ikke var noen systematiske forskjeller i hvilken brukere fikk hvilken av de to designene. Hvis randomiseringen er skjev, kan en lav *p*-verdi skyldes systematiske forskjeller i brukerne, ikke at design B faktisk er bedre.

Oppgave 5: Regresjon

En liten iskrembod registrerer døgnmiddeltemperatur `x` (målt i °C) og antall solgte iskremkjeks `y` den dagen. Eieren tror at sammenhengen er tilnærmet lineær for typiske sommerdager:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

der ε er uavhengig normalfordelt tilfeldig feilled med forventning 0 og standardavvik σ.

5a. Skriv en Python-funksjon med signaturen

```
def simulate_dataset(beta0, beta1, sigma, n_days, T_min, T_max):
    ...
```

som simulerer ett syntetisk datasett og returnerer en tuppel `(x_list, y_list)` bestående av to lister av lik lengde `n_days`. Temperaturene `x_list` skal trekkes uavhengig fra en uniform fordeling på `[T_min, T_max]`, og salgstallene `y_list` skal genereres i henhold til modellen over med normalfordelt støy. Tving salgstallene til å være flyttall; ikke avrund eller kutt.

Løsningsforslag

5a.

```
import random
from scipy.stats import norm

def simulate_dataset(beta0, beta1, sigma, n_days, T_min, T_max):

    x_list = [random.uniform(T_min, T_max) for _ in range(n_days)]
    y_list = [beta0 + beta1 * x + norm.rvs(scale=sigma) for x in x_list]
    return x_list, y_list
```

Oppgave 5: Regresjon

5b. Anta at du har brukt funksjoner (f.eks. fra `statsmodels` modulen) til å tilpasse en lineær regresjon til et syntestisk datasett generert med

```
simulate_dataset(20, 4, 6, 90, 15, 30)
```

og at denne funksjonen har rapportert estimerte koeffisienter `b0 = 17,2` og `b1 = 4,1`.

(i) Forklar med egne ord hva tallet `b1 = 4,1` betyr i denne modellen.

(ii) Forklar med egne ord hva tallet `b0 = 17,2` betyr, og hvorfor tolkningen av skjæringspunktet kan være mindre interessant i denne konkrete situasjonen.

Løsningsforslag

5b.

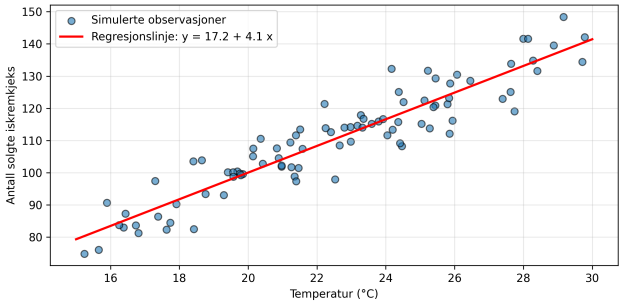
Modellen er $\widehat{y} = b_0 + b_1 x$.

(i) Stigningstallet b_1 betyr at når døgnmiddeltemperaturen øker med 1 grad Celcius, predikerer modellen i gjennomsnitt omtrent 4,1 flere solgte iskremkjeks, alt annet likt.

(ii) b_1 er skjæringspunktet. Tolkningen her er at ved 0 grader, så predikerer modellen i gjennomsnitt ca. 17,2 solgte iskremkjeks. Tolkningen av skjæringspunktet er mindre interessant fordi dataene er generert for sommerdager med døgntemperatur i intervallet $[15, 30]$, så 0 grader ligger langt utenfor det området der modellen er basert på observasjoner. Vi bør være forsiktige med å gi skjæringspunktet en praktisk betydning når det ligger langt utenfor det relevante x-området.

Oppgave 5: Regresjon

5c. Figuren under viser et spredningsplott av det syntetiske datasettet fra 5b, med regresjonslinjen med `b0 = 17,2` og `b1 = 4,1` overlagret.



(i) Er disse estimerte koeffisientene i tråd med hva du ville forventet ut fra simuleringsmetoden gitt ovenfor? Gi en kort forklaring, inkludert hvorfor vi ikke bør forvente at estimatene treffer nøyaktig 20 og 4.

(ii) Året etter registrerer eieren et reelt (ikke simulert) datasett og kjører samme tilpasningsprosedyre. Han får nå `b0 = 18, b1 = 4.6`. Hvilken informasjon trenger du i tillegg til disse estimatene for å vurdere om resultatene er konsistente eller inkonsistente med den syntetiske modellen $y = 20 + 4x + \varepsilon$? Gi en kort begrunnelse.

Løsningsforslag

5c.

(i) Vi har simulert 90 datapunkter fra modellen $y = 20 + 4x + \varepsilon$, med $\varepsilon \sim N(0, 6^2)$ og x uniform på $[15, 30]$. Vi ville i denne sitasjonen forvente estimater b_0 og b_1 som ligger nærme de sanne verdiene på henholdsvis 20 og 4, men vi har et begrenset datasett hvor datapunktene varierer rundt linjen (med standardavvik på 6), så estimatene vil variere noe.

(ii) For å vurdere om disse resultatene er konsistente med modellen trenger vi informasjon om usikkerheten i estimatene, f.eks. standardfeil for b_0 og b_1, konfidensintervaller for β_0 og β_1, eller p-verdier for testene $H_0 : \beta_0 = 20$ mot $H_a : \beta_0 \neq 20$ og $H_0 : \beta_1 = 4$ mot $H_a : \beta_1 \neq 4$. Med denne informasjonen kan vi formelt vurdere om resultatene er konsistente med modellen.

Oppgave 5: Regresjon

5d. Eieren vurderer nå en vinterdag med temperatur på 5 °C og bruker regresjonslinjen fra figuren (med `b0 = 17,2` og `b1 = 4,1`) til å predikere salget. Den predikerte verdien er omtrent `17,2 + 4,1 × 5 = 38` iskremkjeks. Gi to ulike innvendinger mot å stole på denne prediksjonen, basert på det du vet om regresjon fra dette kurset.

Løsningsforslag

5d. Prediksjonen er basert på ekstrapolasjon langt utenfor dataintervallet og kundeadferd kan være helt annerledes på vinteren. Regresjonsmodellen bruker bare temperatur som forklaringsvariabel, men det kan være andre faktorer har stor betydning, spesielt på en kald vinterdag. Denne modellen gjelder ikke nødvendigvis under helt andre forhold.

Oppgave 5: Regresjon

5e. Anta at feilleddet ε faktisk avhenger av temperaturen: variasjonen fra dag til dag er mindre på kjølige dager, mens variasjonen er mye større på varme dager.

(i) Vil en vanlig lineær regresjonsrutine likevel kunne gi omtrent korrekte estimater av β_0 og β_1? Forklar kort.

(ii) Ville de standard antakelsene bak ordinær lineær regresjon fortsatt vært oppfylt? Gi en kort begrunnelse.

Løsningsforslag

5e.

(i) Hvis feilleddene ε fortsatt har forventning 0 for alle verdier av x, gjelder fortsatt at forventningen i modellen er korrekt spesifisert: $\mathbb{E}[Y \mid x] = \beta_0 + \beta_1 x$. Estimatene av β_0 og β_1 vil derfor fortsatt være omtrent forventningsrette (og konsistente).

(ii) En av standard antakelsene bak lineær regresjon er at variansen til feilleddet er den samme for alle verdier av x. Brudd på denne antakelsen påvirker kvaliteten på usikkerhetsestimatene (standardfeil, konfidensintervaller, tester) som ikke er pålitelige lenger. De andre antakelsene om lineær sammenheng mellom respons og forklaringsvariabler, normalfordelte feilledd og uavhengige observasjoner kan fortsett gjelde.