

Forbedringer til reproduktionsindikatorer for løbende inseminerings og drægtighedsopfølgningen i Kvægbesætninger implementeret i Kritiske Målepunkter



Opsummering af forbedringer udarbejdet i løbet af 2013

Der har i løbet af 2013 været arbejdet på at forbedre de nuværende reproduktionsindikatorer i KMP. Det har primært været udfordringer på bedrifter med under 100 køer og på bedrifter der i gentagne uger har insemineret på alle brunster (ins% på 100) eller gentagne uger i træk ikke har insemineret (ins% på 0). Her kunne forekomme kurver der i for lang tid holdt sig på en meget høj eller lav værdi eller at kurverne gav fejl og ikke kunne tegnes. Nedenstående beskriver den løsning, som er fremkommet.

25/11-13

Baggrund og ændringer

Som opfølgning på arbejdet, som vi har haft – kan jeg hermed opsummere, at jeg mener, at vi har fået klaret de tre nedenstående problemstillinger:

3. Brud i den fremadrettede M/C-estimation
4. For mange fejlkoder 103 (m eller C bliver uendelig)
5. Øjensynlig for langsom adaption til data

Ad 3: jeg har ændret i R-koden, så R genberegner fra starten af hver gang. Det betyder, at R nu ikke har brug for feltet AfterWeek i TReproInRAdmin. Og ikke mindst at vi fremover undgår at gamle brud på den fremadrettede M/C-estimation bliver hængende i tabellerne. Det betyder så også en marginalt længere beregningstid i R. Til gengæld har jeg sørget for at R kun lægger resultater tilbage i DWH-tabel, såfremt der er tale om en ændret datalinie (målt på 8 decimaler).

Desuden har jeg ændret det sådan at Smooth-beregning *altid* går 53 datalinier (dvs. uger) bagud i tid, fordi det er dette antal uger, som vises på KMP's side 2-grafer. Derfor er feltet SmoothWeek ikke længere nødvendig i TReproInRAdmin.

Ad 4+5: Her har jeg med Anders Ringgaards hjælp fået ændret i M/C-estimation, således at vi ikke længere får fejlkode 103 og at m ikke løber af sporet. Desuden har jeg tjekket at den nedre grænse i 98% af tilfældene forbliver næsten den samme (dvs. højst en forskel på ± 1) – uanset om vi bruger den metode, som allerede ligger i DWH, eller vi bruger den nye Taylor-metode. Men selvfølgelig vil Taylor-metoden nu give en nedre grænse, hvor den gamle metode gav ukendt alarmgrænse. (Se beskrivelse herunder) ☺

Jeg har testet på ca. 14.000 datablokke i DWH: at m opfører sig pænt, og at vi i 98% tilfælde har samme alarmgrænse (inden for ± 1). Dette har jeg gjort ved kørsel på egen pc, men på data fra DWH.

Valget af C0, m.limit og C.limit værdier er ikke længere så vigtige – specielt fordi limit-værdier ikke længere nås. Derfor mener jeg blot at vi kan beholde de værdier, som er anført nedenfor:

Initialværdi af $C = C_0 = 0.5$

Grænse for $m = 30$

Grænse for $C = 1E+12$

Implentering i DWH

Jeg mener derfor, at vi er klar til at implementere de nye beregninger i DWH. Dvs. først en test i Medio med den kode, som jeg har liggende. Det er en modificeret udgave af R-koden fra DWH. Den skulle gerne kun være ændret således, at den er klar til at blive lagt ind i DWH-kørslen (blot skal værdien `Test.No.Write` ændres til `FALSE`).

Jeg har også ændret og testet R-koden, som beregner den nedre alarmgrænse (i `KVDSAReproAlarm.R`). Den bruger nu også Taylor-metoden. Her kræves dog ingen ændring af datatabel.

Jeg har allerede på min maskine – men på data fra DWH-prod. – testet:

- at R ikke går ned
- at R danner et meningsfuldt output, og det er sammenligneligt (og ofte bedre), end det, som vi allerede har i DWH.

Herefter mener jeg, at vi skal følge i DWH:

- fjerne og tilføje ønskede felter i `TReproInRAdmin` (se nedenfor)
- teste at R-koden kører rigtigt, uden nedbrud, og som forventet på DWH-medio

9/11-13

Undersøgelse af Taylor-DGLM

Som opfølgning har jeg nu kigget på, hvordan konfidensgrænserne for forventet antal ser ud, når vi sammenligner Taylor-DGLM med den algoritme som gennem længere tid har kørt i DataWareHouset (DWH). Det er mit håb, at især nedre grænse er fuldt ud sammenlignelig, sådan at grænserne i langt de fleste tilfælde ikke vil ændre sig mærkbart, selv om vi ligger Taylor-DGLM ind i DWH-kørslen.

Jeg har derfor sat min computer til at køre 6.870 datablokke igennem (kun for insemineringer, `InsDrgt=1`, da det kun er for insemineringer, at vi bruger konfidensintervallerne). For hver datablok har jeg kigget på de sidste 20 ugers grænseværdier. Dvs. potentielt op mod $6870 \cdot 20 = 137.440$ grænseværdier. For hver af disse har jeg sammenlignet den grænse, som er angivet i DWH nu, og den grænseværdi som kommer fra Taylor-DGLM.

Jeg har dog lavet et par ændringer i din kode, Anders.

1. Jeg har ændret betingelsen for at bruge binomial i stedet for BetaBinomial marginalt, for jeg har skrevet
`if (alpha <= 0 | beta <= 0)` i stedet for `if (alpha < 0 | beta < 0)`
dvs. at tilfældet med α eller β lig nul skal håndteres som binomial, fordi jeg mener, at det vil gå galt i BetaBinomial, hvis vi beregner $\lgamma(0)$. Det er ret usandsynligt at ændringen overhovedet betyder noget, men jeg vil gerne være sikker på at R ikke går ned.
2. Jeg lægger *ikke* +1 til resultatet, når jeg beregner den øvre konfidensgrænse.
Jeg bruger nemlig en ny funktion for `QuantileBetaBinomial`. På samme måde som i `qbinom` returnerer denne funktion en fraktil som opfylder: The quantile is defined as the smallest value x such that $F(x) \geq p$, where F is the distribution function, jf. R-dokumentationen for `qbinom`. Dermed bliver den

returnerede fraktil lig med det antal, som er acceptabel inden for konfidensgrænserne. Dette gælder både for den øvre og nedre grænse. Mine nærmere overvejelser om fraktiler og alarmgrænser er angivet i vedhæftede kopi af OneNote-notat.

3. Desuden bruger jeg hhv. 5% og 95% grænse, for det har man for lang tid siden ønsket til Repro-KMP

Ved sammenligningen af de 137.400 grænseværdier er der ca. 10%, hvor grænseværdien ikke kunne beregnes i DWH. Det skyldes det kendte problem med at Estimation-M/C løber af sporet.

Men blandt de tilbageværende ca. 124.000 grænseværdier giver 76% af dem samme resultat for den nedre grænseværdi, uanset om vi får den fra Taylor-DGLM eller læser den i DWH. Og i yderligere 22% er der kun en difference på +1 eller -1. Dvs. at i 98% af tilfældene er der maksimalt en forskel på 1!

For den øvre grænseværdi har vi samme resultat fra DWH og Taylor-DGLM i 68% af tilfældene. Og en difference for +1 (DWH-grænse størst) i 5% af tilfældene, og en difference på -1 (Taylor-grænse størst) i 24% af tilfældene. Dvs. at i alt 97% af tilfældene er der maksimalt en forskel på 1 for den øvre grænse.

I alt mener jeg at det er ganske pænt, og taler for at vi sagtens kan benytte os af Taylor-DGLM i stedet for den nuværende algoritme i DWH. Specielt synes Taylor-DGLM at være brugbar, når vi ser på at denne algoritme overfører sig meget pænere i forhold til beregningen af m.