Check Update Feature of Breed Info

Peter von Rohr

2021-02-10

Disclaimer

The update feature of the page that allows to specify breed-specific information is documented and verified.

Experiments and Tests

The following values were entered on the Breed-Info Page for the test cattle breed



Welcome to the update page. You are updating information for the breed:

Cattle Test Breed 2020-11-12 (ctb20201112)

General information about the breed:

Does the breed have a cultural value
Does the cultural value of the breed decreased in the recent past yes no
Please give the approximate number of farms 100 Please give the approximate number of farms 5 years ago 150
Does the breed have cryo-conserved semen yes no
Does the breed have a cryo-conservation management plan? yes no
Update breed information

The following entries were recorded in the logfile ''

```
| Topr = quagadmin@fagr ~ = ssh quagadmin@fagr. q=mon.ch = 154×53 | Topr = quagadmin@fagr. q=quagadmin@fagr. q
```

The result on the main web-site is then shown as

Breed name	Last year of data	Number individals last GI	Pedigree compl. 6th gen, last GI	Average inbreeding last GI	Effective pop size (Ne_DeltaFp)	Trend males last 5 years	Trend females last 5 years	Pedig Index (0-1)	Introg Index	Geog index (km)	BAS Index (0-1)	Cryo-cons score (0-1)	Global Index (0-1)	Delete Breed	More Info
ctb20201112 (cattle)	2008	289	66.2	0.008	256 ■	-8.96	-13.14	0.55	0.078 -	53.86	0.631	1 •	0.678	delete	more

Index Computation

The following sql-statements are used when computing the different indices

psql -U postgres -c "select * from thres_weight where owner='zwsgnm' and crit_type='demo' and species=' Show the table with the weights

```
psql -U postgres -d GenMon_CH -c "SELECT * FROM pg_catalog.pg_tables where tablename like '%thres%'"

schemaname | tablename | tableowner | tablespace | hasindexes | hasrules | hastriggers | rowsecuri

------

public | thres_weight | geome_admin | | f | f | f | f |

(1 row)
```

Select entries in the table of weights

```
| -5 |
                       0 |
                           0.15 | demo
                                            | zwsgnm | cattle
trend_males
                       0 | 0.15 | demo
trend_females | -5 |
                                           | zwsgnm | cattle
            | 0.1 | 0.03 | 0.15 | demo
                                           | zwsgnm | cattle
             | 50 | 250 |
                            0.4 | demo
                                           | zwsgnm | cattle
             | 87 | 97 | 0.15 | demo
                                           | zwsgnm | cattle
ped_compl
(5 rows)
```

The selection in the first loop-round

```
psql -U postgres -d GenMon_CH -c "select t.weight, t.t1, t.t2 from thres_weight t
     where crit_type='demo' and t.criteria='trend_males' and owner='zwsgnm' and species='cattle'
     union
```

```
select t.weight, t.t1, t.t2 from thres_weight t
where crit_type='demo' and t.criteria='trend_males' and owner='zwsgnm' and species='default'
and not exists (select t.weight, t.t1, t.t2 from thres_weight t
   where crit_type='demo' and t.criteria='trend_males' and owner='zwsgnm' and species='cattle')"
```

Summary select

 $psql - U \ postgres - d \ GenMon_CH - c \ "select \ trend_males \ from \ summary \ where \ breed_id = 11 \ and \ trend_males \ id \ summary \ breed_id = 11 \ and \ trend_males \ summary \ breed_id = 11 \ and \ summary \ breed_id = 11 \ and \ summary \ breed_i$

Der globale Index wird erneuert. In 'GenAnimalUpdate.php' wird der globale Index und der 'demo'-Index neu berechnet. Der 'demo'-Index ändert aber nicht mit Änderungen von Breed-Info. Gemäss den Eingaben in der Tabelle 'thres weight' mit 'crit type' = 'demo' ist der 'demo'-Index nur abhängig von

- trend males
- trend females
- avg inb
- ne
- ped compl

Diese folgenden Grössen werden auf der Breed-Info-Seite (UpdateBreed.php) geändert

- cultural value
- cultural value trend
- number farm
- number farm past
- frozen_semen
- cryo plan

BAS-Index

Wie wird der BAS-Index berechnet? In welchem Feld wird BAS-Index abgelegt

Aufgrund des Wertes ist es das Feld 'index_socio_eco'. Dieses Feld wird im Postprocessing der PopRep-Resultate und über die Funktion 'IndexSocioEcPLZ' in FunctionsCalcIndex.php erneuert.

```
grep -n -r -i 'update summary set index_socio_eco' /var/www/html/genmon-ch/
/var/www/html/genmon-ch/FunctionsCalcIndex.php:210: $sql_breed_index1= "UPDATE summary SET index_so
/var/www/html/genmon-ch/PopRep.php:353:$sql_breed_summary[] = "UPDATE summary SET index_socio_eco =
/var/www/html/genmon-ch/PostprocessPopRep.php:353:$sql_breed_summary[] = "UPDATE summary SET index_soci
```

Die Funktion 'IndexSocioEcPLZ' wird nur in 'ChangeWeightSocioEco.php' bei der Änderung der Gewichte und in 'GenOFS.php' bei der Verarbeitung von neuen Daten vom BFS aufgerufen.

Die Faktoren, welche den BAS-Index beeinflussen sind:

```
psql -U postgres -d GenMon_CH -c "select * from thres_weight where crit_type = 'SocioEco' and owner = 'socioEco' a
```

percent_less_19	3	10	0.05 SocioEco	zwsgnm default
percent_more_65	20	4	0.15 SocioEco	zwsgnm default
unemployment_rate	5	2	0 SocioEco	zwsgnm default
demog_balance	0	3	0.05 SocioEco	zwsgnm default

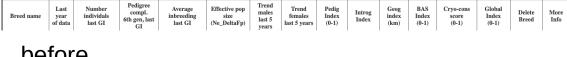
Socio-Econ Index

Computation of IndexSocioEcPLZ must be done in UpdateBreed.php. Determine the year with the following select

psql -U postgres -d GenMon_CH -c "SELECT cast(substring(table_name, 13,4) as integer) as table_name2 FR table_name2 2014

(1 row)

After the re-computation of the socio-econ index



before



after



Check the values that were set

psql -U postgres -d GenMon_CH -c "(SELECT round(cast(sum(a.num_ind_lastGI*b.index_socioec)/sum(a.num_in FROM breed11_inb_plz a, plzo_plz b WHERE a.plz=b.plz)" round

0.631 (1 row)

The above value consists of

psql -U postgres -d GenMon_CH -c "SELECT * FROM breed11_inb_plz a, plzo_plz b

WHERE a.plz=b.plz"

psql -U postgres -d GenMon_CH -c "SELECT *

FROM breed11_inb_plz

where mean inb lastgi is not null"

plz	mean_inb_lastgi	max_inb_lastgi	num_ind_lastgi	mean_inb_gen_lastgi	mean_introgr_lastgi
	+	+	+	+	
1041	1 0	0	1		0.06
1072	1	1 0	1	1	0.03
1180	1 0	1 0	1		0.15
1186	0.0184402	0.0184402	l 1	 	0.06

1124 | 0 | 0 | 1 | 0.02

the above shows the values for 'mean_inb_lastgi', 'max_inb_lastgi', 'num_ind_lastgi', 'mean_inb_gen_lastgi' and 'mean_introgr_lastgi across the plz values. We note that mean_inb_gen_lastgi is null for all records in breed11_inb_plz.

The description of the table plzo_plz is obtained by

psql -U geome_admin -d GenMon_CH -c "select column_name, data_type, character_maximum_length, column_de from INFORMATION_SCHEMA.COLUMNS where table_name = 'plzo_plz';"

column_name	data_type	character_maximum_length	column_default	is_nullable
plz	integer	 	 	NO
geom	USER-DEFINED			YES
num_ofs	integer			YES
mean_inb_lastgi	real			YES
max_inb_lastgi	real			YES
num_ind_lastgi	integer			YES
percent_change_wsl	real			YES
demog_balance	real			YES
median_income	real			YES
unemployment_rate	real			YES
percent_farmer	real			YES
percent_grazing_surface	real			YES
percent_less_19	real			YES
percent_more_65	real			YES
evol_job_primary_sector	real			YES
index_socioec	real			YES
mean_introgr_lastgi	real			YES
mean_inb_gen_lastgi	real			YES
(18 rows)				

Some interesting fields of plzo_plz

psql -U postgres -d GenMon_CH -c "SELECT plz, num_ofs, index_socioec FROM plzo_plz"

-		_		index_socioec
6574 4123 1307 4582	1 1 1	5398 2762 5487 2446	 	0.38 0.477 0.741 0.434

. . .

Doing the join between 'breed11_inb_plz' and 'plzo_plz' with the fields selected, we get

psql -U postgres -d GenMon_CH -c "select a.plz, a.num_ind_lastGI, b.index_socioec, a.num_ind_lastGI*b.index_socioec, a.num_ind_lastGI*b.index_

WHERE a.plz=b.plz"

plz	num_ind_lastgi	index_socioec	weight_index
6574	 	0.38	
4123		0.477	1
1307		0.741	1
4582		0.434	1
7223		0.671	

```
0.57 |
8590 I
 1885 I
                       1
                                  0.498 I
9050
                                  0.728 |
                                  0.395 |
5703 |
                       1791 |
                     1 |
                                  0.56 | 0.560000002384186
8180 |
                                  0.534 |
                       With the aggregation function
psql -U postgres -d GenMon_CH -c "(SELECT round(cast(sum(a.num_ind_lastGI*b.index_socioec)/sum(a.num_in
FROM breed11_inb_plz a, plzo_plz b
WHERE a.plz=b.plz)"
0.631
(1 row)
The sql-statement that has no consequences on the values in the summary table:
psql -U postgres -d GenMon_CH -c "(SELECT round(cast((aa.plz_value+bb.cult+cc.farm) as numeric),2)
FROM (SELECT sum(a.index_socio_eco*b.weight) as plz_value
FROM summary a, thres_weight b
WHERE b.crit_type='SocioEco'
AND b.criteria NOT LIKE 'breed%'
AND b.owner='zwsgnm'
AND a.breed_id=11) aa,
(SELECT d.weight*c.breed_cultural_value as cult
FROM summary c, thres_weight d
WHERE d.criteria='breed_cultural_value'
AND d.owner='zwsgnm'
AND c.breed_id=11) bb,
(SELECT f.weight*e.breed_num_farms_trend as farm
FROM summary e, thres_weight f
WHERE f.criteria='breed_num_farms_trend'
AND f.owner='zwsgnm'
AND e.breed_id=11) cc)"
round
 -1.11
(1 row)
```

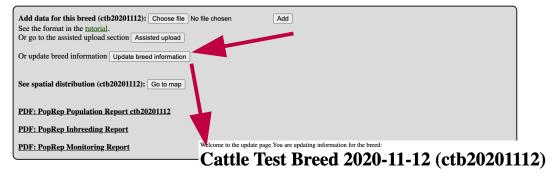
Summary on Update of Breed Info

The page to update breed info shown in the screen-shot below allows the user to change the following values

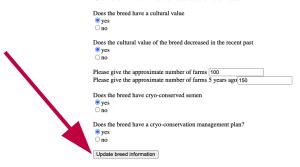
- cultural value
- cultural_value_trend
- number_farm
- number_farm_past
- frozen semen
- cryo_plan

GenMon-CH

Cattle Test Breed 2020-11-12 (ctb20201112)



General information about the breed:



Once the 'Update-Button' is pressed, two indices (Pedig-Index and Global Index) are re-computed. But because the values changed on the update-breed-info page do not influence the Pedig-Index, this value will not change. The Pedig-Index only depends on

- trend_males
- trend_females
- avg_inb
- ne
- ped_compl

Hence one could remove the re-computation of the Pedig-Index.