

CMED Variable Summary

We have made some significant changes in reporting derived CMED (Concomitant Medications) variables compared to the previous datasets submitted to NIDDK.

1. Previously, medications were assigned to one of 15 medication categories (e.g., Oral Narcotics, Alpha Blockers, SSRIs, etc.) , and each of these categories was then assigned to a pain category level (1=Peripheral, 2=Central, 3=Opioids) Thus, at a given visit, we reported the following:
 - a. Indicator variables for the 15 medication categories
 - b. Maximum pain category based on the medication categories endorsed
2. Upon review, we recognized some shortcomings with this approach:
 - a. The list of medication categories did not encompass all medications reported
 - b. The list of medications assigned to each category was not necessarily complete
 - c. It did not allow room to distinguish between medications taken for urinary/pelvic pain vs. those taken for other purposes in reporting maximum pain category
3. Thus, we made the following changes:
 - a. We no longer use the medication category variables
 - b. We compiled a list of each unique medication reported, and an MD with expertise in treating pain assigned each individual medication to a pain category level:
 - 0 = None
 - 1 = Peripheral
 - 2 = Central
 - 3 = Opioid
 - c. This list of pain category assignments is provided as SAS dataset `cmed_drugs_paincat.sas7bdat` and is referenced in the SAS code that follows as working dataset `medpaincats1`
 - d. Maximum pain category for each person/visit is now reported as two separate variables:
 - i. `cmed_anypain_maxpaincat` = The highest pain category among all meds taken at a given visit
 - ii. `cmed_ucppspain_maxpaincat` = The highest pain category among those meds taken at a given visit for which the participant indicated the medication was taken for Urologic/Pelvic pain

SAS Code

```
/* assign the pain categories to the drugs in the baseline and longitudinal datasets, and determine the  
max-pain-cat for all reported meds and those reported for uro/pelv pain for each person/visit */
```

```
/* merge the Baseline cmed data with the pain-category assignments */
```

```
data blmeds2;  
merge blmeds (in=a) medpaincats1 (in=b);  
by drug_code;  
if a and b;  
run;
```

```
proc sort data=blmeds2;  
by pid;  
run;
```

```
data blmeds_h1;  
set blmeds2;  
by pid ;  
retain cmed_anypain_maxpaincat cmed_ucppspain_maxpaincat;
```

```
if first.pid then do;  
    cmed_anypain_maxpaincat = 0;  
    cmed_ucppspain_maxpaincat = 0;  
end;
```

```
if PainCat_3level gt cmed_anypain_maxpaincat then cmed_anypain_maxpaincat = PainCat_3level;  
if med_for_uropelvic_pain eq 1 and PainCat_3level gt cmed_ucppspain_maxpaincat then  
    cmed_ucppspain_maxpaincat = PainCat_3level;
```

```
if last.pid then do;  
    output;  
    cmed_anypain_maxpaincat = .;  
    cmed_ucppspain_maxpaincat = .;  
end;
```

```
label cmed_anypain_maxpaincat = "CMED Max Pain Category - all meds taken";  
label cmed_ucppspain_maxpaincat = "CMED Max Pain Category - meds taken for Uro/Pelvic Pain";  
format cmed_anypain_maxpaincat cmed_ucppspain_maxpaincat medcatf.;  
keep pid cmed_anypain_maxpaincat cmed_ucppspain_maxpaincat;  
run;
```

```
proc sort data=blmeds_h1 nodupkey;  
by pid;  
run;
```

```
/* create other summary variables */
```

```
proc freq data=blmeds2a noprint;  
tables pid/out=pidcounts_bl (keep=pid count);  
run;
```

```
proc freq data=blmeds2a noprint;  
where med_for_uropelvic_pain eq 1;  
tables pid/out=uropelv_counts_bl (keep=pid count);  
run;
```

```
data pidcounts_bl1;  
set pidcounts_bl;  
rename count = cmed_num_meds;  
run;
```

```
proc sort data=pidcounts_bl1 nodupkey;  
by pid;  
run;
```

```
data uropelv_counts_bl1;  
set uropelv_counts_bl;  
rename count = cmed_num_uropelv_meds;  
run;
```

```
proc sort data=uropelv_counts_bl1 nodupkey;  
by pid;  
run;
```

```
data counts_bl;  
merge m1pids (in=a) uropelv_counts_bl1 (in=b) pidcounts_bl1 (in=c);  
by pid;  
if a;  
if b then cmed_any_uropelv_meds = 1;  
else do;  
    cmed_any_uropelv_meds = 0;  
    cmed_num_uropelv_meds = 0;  
end;
```

```
if c then cmed_any_meds = 1;  
else do;  
    cmed_any_meds = 0;  
    cmed_num_meds = 0;  
end;  
run;
```

```

proc sort data=counts_bl nodupkey;
by pid;
run;

data cmed_bl_comb;
merge counts_bl (in=a) blmeds_h1 (in=b);
by pid;
if a;

if cmed_any_meds eq 0 then do;
    cmed_anypain_maxpaincat = 0;
    cmed_ucppspain_maxpaincat = 0;
end;

run;

```

/ Repeat for Longitudinal */*

```

data longmeds2;
merge longmeds (in=a) medpaincats1 (in=b);
by drug_code;
if a and b;
run;

proc sort data=longmeds2a;
by pid vnum;
run;

data longmeds_h1;
set longmeds2;
by pid vnum;
retain cmed_anypain_maxpaincat cmed_ucppspain_maxpaincat;

if first.vnum then do;
    cmed_anypain_maxpaincat = 0;
    cmed_ucppspain_maxpaincat = 0;
end;

if PainCat_3level gt cmed_anypain_maxpaincat then cmed_anypain_maxpaincat = PainCat_3level;
if med_for_uropelvic_pain eq 1 and PainCat_3level gt cmed_ucppspain_maxpaincat then
    cmed_ucppspain_maxpaincat = PainCat_3level;

if last.vnum then do;
    output;
    cmed_anypain_maxpaincat = .;
    cmed_ucppspain_maxpaincat = .;
end;

```

```
label cmed_anypain_maxpaincat = "CMED Max Pain Category - all meds taken";  
label cmed_ucppspain_maxpaincat = "CMED Max Pain Category - meds taken for Uro/Pelvic Pain";  
format cmed_anypain_maxpaincat cmed_ucppspain_maxpaincat medcatf.;
```

```
keep pid vnum cmed_anypain_maxpaincat cmed_ucppspain_maxpaincat;  
run;
```

```
proc sort data=longmeds_h1 nodupkey;  
by pid vnum;  
run;
```

```
/* now create other summary variables */
```

```
proc freq data=longmeds2a noprint;  
tables pid*vnum/out=pidcounts_long (keep=pid vnum count);  
run;
```

```
proc freq data=longmeds2a noprint;  
where med_for_uropelvic_pain eq 1;  
tables pid*vnum/out=uropelv_counts_long (keep=pid vnum count);  
run;
```

```
data pidcounts_long1;  
set pidcounts_long;  
rename count = cmed_num_meds;  
run;
```

```
proc sort data=pidcounts_long1 nodupkey;  
by pid vnum;  
run;
```

```
data uropelv_counts_long1;  
set uropelv_counts_long;  
rename count = cmed_num_uropelv_meds;  
run;
```

```
proc sort data=uropelv_counts_long1 nodupkey;  
by pid vnum;  
run;
```

```

data counts_long;
merge long_pidvnums (in=a) uropelv_counts_long1 (in=b) pidcounts_long1 (in=c);
by pid vnum;
if a;
if b then cmed_any_uropelv_meds = 1;
else do;
    cmed_any_uropelv_meds = 0;
    cmed_num_uropelv_meds = 0;
end;

if c then cmed_any_meds = 1;
else do;
    cmed_any_meds = 0;
    cmed_num_meds = 0;
end;
run;

proc sort data=counts_long nodupkey;
by pid vnum;
run;

data cmed_long_comb;
merge counts_long (in=a) longmeds_h1 (in=b);
by pid vnum;
if a;

if cmed_any_meds eq 0 then do;
    cmed_anypain_maxpaincat = 0;
    cmed_ucppspain_maxpaincat = 0;
end;

run;

```