Details of changes to .do files:

Launch Programme.do

1. Users can choose between three options in relation to furlough. The three options are:
   - o 'ignore' furlough, focusing on the underlying employment spell
   - o treat furlough as a separate spell
   - o treat furlough as another non-employment status, which is what would happen if you were to run the LW code as it was written prior to furlough

   This choice is enacted by choosing the value of global `furlough_choice`:

   ```
   global furlough_choice            "nofurlough"
   ```

   - The default value "nofurlough" entails that furlough spells will be subsubmed into the underlying employment spell. Furlough will not appear as a separate status. This will also happen if `furlough_status` does not take either of the other two explicit values.
   - There are two alternative choices for `furlough_choice`:
     - o Value "furlough" retains furlough statuses. For those on furlough, `Status` coding will be unusual: the last 2 digits will represent the usual UKHLS furlough status (12 or 13). The other digits will represent the identifiable simultaneously-held employment status (1,2,100 - self-employed, employed, in work with no information about self empl/empl). So when option "furlough" is chosen, status values when furloughed are provided as 112,113,212,213,10012,10013.
     - o Value "noadjust" treats `jbstat` values 12 and 13 as the original LW code (developed prior to Covid) would: any status other than `inlist(jbstat,1,2,100)` is treated as non-employment. Statuses 12 and 13 will appear but no account of these representing furlough is taken.

2. Users can select a value for the minimum age at which they believe that individuals can have a status that is not full-time education, i.e. the earliest age at which a non-education status such as employment will not be treated as an error and discarded. This is chosen by picking a number for `noneducstatus_minage`.

   Detail for `noneducstatus_minage` (Minimum age for non-education status):

   - All observations for a `pidp` in the relevant Wave will be dropped if that `pidp` has a non-education spell starting or ending at an age below this value.
   - Which data are affected by choice of `noneducstatus_minage`? The value is used to select data from `lifemst` files (BHPS Life History) and will impact all datasets using those data.
   - Default value of `noneducstatus_minage`:

     ```
     global noneducstatus_minage          0
     ```

     The default value of 0 means the resulting dataset includes all spells starting or ending after birth. Note that spells starting or ending before a particular age can be dropped outside of dataset creation stage of your research.
   - Suggested alternative values for `noneducstatus_minage`:
     - o 10 would replicate Liam Wright's code in this respect.
     - o Other options include 16, or other values depending on research needs.
   - A red notification message will appear on screen when `noneducation_minage` is used: "You have chosen to retain non education statuses after age `$noneducstatus_minage`".

- Comparison with Liam Wright's (2020) code: LW code drops `pidp-Waves` (all data for an individual within an affected wave) where the raw data record that the individual started or ended a non-education spell before age 10. In contrast, use of `noneducstatus_minage` with default value of 0 will retain all non-education statuses as long as they start after an individual's date of birth.

3. Users can select the extent to which data will be dropped in response to implausible dates.

   Detail for `implausibledates_drop`:

- Default choice for `implausibledates_drop`:
  ```
  global implausibledates_drop     "obs"
  ```
  The default choice of `"obs"` retains the most data. Only the spells ("obs"ervations) with implausible dates will be dropped. Other spells in that `pidp-Wave` will be retained and renumbered.
- The choice of `implausibledates_drop` is implemented in program `prog_implausibledates`, which is used in cleaning all work and life histories (by "Clean Dependent Annual/Non-Dependent Annual/Life History.do" files).
- Alternative choices for `implausibledates_drop`: Any other value than `"obs"` will lead to all data for that pidp-Wave being dropped.
- A red notification message will appear on screen when `implausibledates_drop` is used – if `"obs"` is chosen, or not, respectively:
  - "You have chosen to just drop the implausible-date spells (`countXX' observations) (rather than dropping the whole pidp-Wave history (`countYY' observations))".
  - "You have chosen to drop the whole pidp-Wave history (`countYY' observations) where at least one date is implausible (rather than just dropping the `countXX' implausible-date spells)".
- Comparison with Liam Wright's (2020) code: LW code drops the whole `pidp-Wave` (all data for an individual within an affected wave) if there was an implausible date for that individual in that Wave. Any other value than `"obs"` for `implausibledates_drop` will replicate this.

4. Users can choose whether to correct, or drop, `pidp-Waves` with non-chronological year, season-year, or monthly spell start dates. The choice will impact data drawn from `lifemst` files (BHPS Life History).

- Default choice for `nonchron_correct`:
  ```
  global nonchron_correct          "Y"
  ```
  The default choice of `"Y"` makes corrections and retains data.
  - Users should be aware that these corrections are in some cases based on detailed data inspection rather than implementation of broad rules.
- Alternative choices for `nonchron_correct`: Leave blank or choose any other value to drop all observations for a `pidp-Wave` where there is a non-chronological spell start date.
- A red notification message will appear on screen when `nonchron_correct` is used – if `"Y"` is chosen, or not, respectively:
  - "BHPS Life History: You have chosen to deal with non-chronological dates by retaining data where possible through plausible date corrections (nonchron_correct=Y)".
  - "BHPS Life History: You have chosen to deal with non-chronological dates by deleting all data for that pidp in the affected Wave (nonchron_correct!=Y)".

- Choosing value “Y” also permits detailed corrections to spell dates during cleaning of BHPS Life History by “Clean Work History.do”.
- Comparison with Liam Wright’s (2020) code: LW code drops the whole `pidp-Wave` (all data for an individual within an affected wave) if there was an non-chronological date for that individual in that Wave. Any other value than `"Y"` will replicate this.

4. A choice retained from the original LW code allows users to choose the gap length across which missing dates will be imputed by choosing the mid-point between plausible lower and upper date bounds.
   - LW suggests a value of 6 (months).
   - Testing indicates that the choice of this value makes very little difference to labour market dynamics.

5. Users can select whether to run the code quietly or noisily. If “QUIETLY” is chosen, the screen will still show (in red) many displayed messages – all of which are only for information and do not require action.

Create Programs.do

Refer to changes to individual programs.

Interview Grid.do

this file is essentially unchanged from LW. A minor change is made involving a variable name: BHPS used to record FT/PT categorisation in `jbft`, and UKHLS in `jbft_dv` , but now both BHPS and UKHLS use `jbft_dv`.

UKHLS Education History.do

1. A correction is made to a value of `Missing_StartDate` dummy in one small part. The impact is to (correctly) impute start month as December if missing start month but start year present and less than end year, and imposing start month as January if start year and end year are the same and the spell ends in January that year.
2. A more substantive change involves the retention of observations where an education spell starts in the same or an earlier year than the recorded education spell end. These observations are retained because in focusing on labour market histories, what matters is the date left full time education, and it is not necessary to obtain a consistent history of all education spells. Cases with education start date after interview date are (still) dropped. The retention of education spell end dates is potentially valuable in helping increase observations on labour market spells. The effect is to increase the number of observations in “UKHLS Education History.dta” by about 270 (2%).

BHPS Education History.do

1. This .do file is revised to prioritise education end dates. This entails not dropping data on the basis of spell start dates (and their comparison with spell end or interview dates).
2. The resulting dataset maximises information on date left full-time education, so that when merged with work history data the maximum amount of work histories are retained. The resulting dataset is not designed for analysing education spells
3. To reiterate, the aim is to maximise retained observations on labour market history. The effect is to increase the number of observations in “BHPS Education History.dta” by about 260 (5%).

Interview Grid.do

This file is essentially unchanged from LW.

FTE Variables – Collect.do

1. It is confirmed (as of Wave 12 [30]) that the correction to ajbstat (Wave 1) is sill needed.
2. `scend_dv` is used here in several places if the alternative is discarding information for that observation. For example, `scend_dv` gives information on school end date for some of those for whom the `school` variable otherwise records them as “Never went to school”. This strategy is open to criticism; as LW notes, it is unclear how the information behind derived variable `scend_dv` is elicited. (Rather than use `scend_dv`, LW code sets school end date to interview date, which is done here for those without `scend_dv` information.)
3. `feend_dv` is likewise used.
4. Education end month is set to June if missing, rather than September or July as at places in LW.
5. As expected, the use of `scend_dv` and `feend_dv` rather than `IntDate` lowers mean education end dates, which should entail less overlap with (and consequent discarding of) labour market histories.

FTE Variables – Clean.do

1. This file includes minor changes to variable selection and a minor reorganisation in relation to data checks.
2. For the same number of individuals, the changes to FTE Variables files result in a small rise in FTE end dates (by about 200, or 0.5%).

UKHLS Initial Job.do

1. Code is altered to incorporate furlough `Status` and `End Reason` variables.
2. Statuses 12 and 13 are retained here and adjusted after these data are merged in “UKHLS Annual History_JCS.do”.
3. `jbstat` and `jbsemp` are kept, not dropped, so that they can be used after the merge to convert reported furlough statuses into the relevant underlying employment status.

UKHLS Annual History.do

1. LW code was written before the pandemic, so could not antiipate the introduction from Wave 11 [29] of added response options for `jbstat` (labour market status0 "12. Furlough" AND "13. Temporarily laid off/short time working" (“furlough spells”).
2. 3 new variants of the “UKHLS Annual History.do” file are provided that differ in their treatment of furlough spells.
   - _orig follows the original coding. This treats a furlough spell like any other non-employment spell.
   - The new unsuffixed coding focuses on the underlying employment status, effectively ignoring (or subsuming) furlough status. This would probably be the version most useful for those interested in traditional macro-labour market classifications (employment, unemployment, non-employment).
   - _F treats a furlough spell as a new spell, but uses new `Status` coding that reflects both the underlying employment spell and furlough status.

   These changes require substantial changes to the code of the “UKHLS Annual History.do” file.

3. Other variables are also altered to reflect pandemic issues:
   - For example, `stendreasX` from Wave 11 [29] adds `stendreas12` =1 if Furloughed. In the original Wave 12 [30] data, `stendreas12` appears ‘out of order’ in relation to other variables and is not captured by the original varlist used in LW to collect variables; revised code ensures that variable is collected.

4. There are changes to spell end reason variables and associated notes.
    - nxtendreas is asked Waves 5-6 [23-24], and nxtendreasX is asked from Wave 7 [25] onwards. [LW version: nxtendreasX questions not asked in Waves 3 and earlier.]
    - stendreas is asked in Waves 2-6 [20-24], and stendreasX is asked from Wave 7 [25] onwards – apart from stendreas12 which is asked from Wave 12 [30].
    - The revised code makes use of stendoth_code, included from Wave 8 [26] onwards. Label stendoth_code indicates that value 24 corresponds to "End of contract". The new syntax recodes value 24 as 5 "Temporary job ended". Values of stendoth_code between 1 and 11 correspond to the equivalent stendreasX, so those values of stendoth_code are recoded into the relevant stendreasX. Those observations are also recorded, without useful reason allocation, in stendreas97, so those cases of for stendreas97 are set to missing.
    - It is no longer necessary to rename reasend97* reasend97_*.

BHPS Annual History – Waves 16-18.do

This file is very similar to LW, though a small amount of code is relocated to the “Clean Dependent Annual History.do” file (to enable that file to work correctly in a loop around variants of UKHLS Annual History data).

Clean Dependent Annual History.do

1. Updated code adjusts for the fact that, for a furloughed individual, furlough status and a continuing underlying employment status are both continuing spells, which has the effect that if furlough spells are recoded as employment spells (as in the unsuffixed version of UKHLS Annual History), the individual is recorded as having two non-ended spells, numbered consecutively. The code drops one of these duplicate unfinished employment spells. This adjustment only has an effect for the unsuffixed version of UKHLS Annual History data (no other annual histories feature duplicates).
2. The file creates a lower bound for the start of furlough spells of March 2000.
3. A small amount of code from the end of the LW version of this file is transferred to other files (UKHLS and BHPS Dependent Annual History files) so that this file can be used to clean both UKHLS and BHPS data (in particular, “UKHLS Annual History_JCS.do” requires variants of that code within a loop).

BHPS Annual History – Waves 1 to 15.do

This file is essentially unchanged from LW.

Clean Non-Dependent Annual History.do

This file is essentially unchanged from LW.

Clean Work History.do

1. For BHPS Life History data only, non-chronological dates are corrected to retain data where possible, if the nonchron_correct "Y" option is chosen.
2. If the “Y” option is not chosen, or if the data are BHPS Annual Histories or UKHLS data, all data for any pidp affected by non-chronological dates (which is almost never the case) is dropped.
3. New code – applying only to a variant of UKHLS Annual History data that focuses on furlough spells – corrects start dates in a small number of cases where a(n imputed) furlough start date is implausible.
4. A specific program prog_lastspellmissing_F to deal with missing last spells is called here for the furlough variant of UKHLS Annual History data. This program prevents furlough spells from

being dragged earlier during the process of imputation, beyond the plausible first start date of furlough spells of March 2020. The program also ensures that imputed furlough statuses are coded in the same way as _F the UKHLS variant. (For the unsuffixed UKHLS Annual History, revised `prog_lastspellmissing` recodes ensures that any imputation involving furlough spells generates the underlying employment status.)

UKHLS Life History.do

This file is essentially unchanged from LW.

BHPS Life History.do

- Preferring to avoid discarding potentially valid and useful information by deleting whole `pidp-Waves` (or `pidp`s) affected by `Spell` or date discrepancies, this file checks, flags and corrects discrepanies in spell ordering and end-of-spell information.
- This file calls the substantially altered (for BHPS life history data) do file "Clean Life History_JCS.do".

Clean Life History.do

- For BHPS life history data, this file responds to user choice of case-by-case correction of non-chronological start dates by using the new .do file "prog_nonchron_BHPS.do" if `nonchron_correct` is set to "Y".

Merge Datasets.do

- Merge ordering changed to Annual>Life>Education.

Apply Labels.do

Label added for `EndReason12`.

Labels.do

Labels changed for `Status`.

prog_nonchron_BHPS.do

This is a new .do file that substitutes for program `prog_nonchron` for BHPS Life History data when the option "Y" is chosen for global `nonchron_correct`. It undertakes close inspection of the data and makes case-by-case correction of non-chronological start dates using a variety of rules. The aim is to retain labour market status history information where possible.