Code Sample [[1]](#footnote-20)

Haoliang Hu [huhaoliang@whu.edu.cn](mailto:huhaoliang@whu.edu.cn)

Dec. 15 2023

## Background

This sample is my code for a data task. It has 5 parts:  
- Part 0 Initialization  
- Part 1 Data Cleaning  
- Part 2 Data Exploration  
- Part 3 Estimation and Causal Inference  
- Part 4 Further Analysis

## 0. Initialization

. clear all  
  
. set more off   
  
. set maxvar 20000  
  
  
. set scheme cleanplots, perm  
(set scheme preference recorded)  
  
.   
. // If the reader wants to replicate the results, he/she just needs to change this global path and put the data in raw\_data file.   
. global path "C:\Users\huhu\Desktop\Code Task\David Chan Data Task\"  
  
. global D "$path\data" //data file  
  
. global Out "$path\out" //result: graph and table  
  
. cd "$D" //set current working directory  
C:\Users\huhu\Desktop\Code Task\David Chan Data Task\data  
  
.   
.   
. import delimited "test\_data.csv",clear  
(encoding automatically selected: ISO-8859-2)  
(8 vars, 8,831 obs)  
  
. save raw\_data.dta, replace  
file raw\_data.dta saved  
  
.   
. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Question 0\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
. // summarize data  
. qui summarize, detail  
  
. qui duplicates list arrive leave  
  
.

Summarize the data. I find this data set contains entries of patient flow. The shift duration typically lasts around 9 to 10 hours, and there are rare situations where the shift lasts only 2 hours. And I checked the duplication of the arrive and leave times of patients and found some observations that may appear to be data entry errors as they have exactly the same arrive and leave times, which may result from coding errors of the ED system. In the following analysis, I take them as true data for convenience.

## 1. Data Cleaning

First I transfer the a.m./p.m. to 24-hours in order to transfer the original datatime format into stata time format. Notably, we should use double here to generate the new stata datatime variable. Finally we get there are patients arriving before their physician’s shift starts and patients discharged after their physician’s shift ends.

. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Question 1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
. // transfer datetime  
. gen shift\_date\_time = date(shift\_date, "DMY")  
  
. format shift\_date\_time %td  
  
.   
. // extract hour and AM/PM indicator from shift\_start and shift\_end  
. // replace noon to 12 pm for conviency  
. qui replace shift\_start="12 p.m." if shift\_start=="noon"  
  
. qui replace shift\_end="12 p.m." if shift\_end=="noon"  
  
.   
. gen hour\_start = real(substr(shift\_start, 1, strpos(shift\_start, " ") - 1))  
  
. gen am\_pm\_start = substr(shift\_start, -4, 4)  
  
.   
. // do the same for shift\_end  
. gen hour\_end = real(substr(shift\_end, 1, strpos(shift\_end, " ") - 1))  
  
. gen am\_pm\_end = substr(shift\_end, -4, 4)  
  
.   
. // convert to 24 hour format  
. qui replace hour\_start = hour\_start + 12 if am\_pm\_start == "p.m." & hour\_start != 12  
  
. qui replace hour\_end = hour\_end + 12 if am\_pm\_end == "p.m." & hour\_end != 12  
  
.   
. // combine data with time  
. gen shift\_start\_data\_time = shift\_date + " " + string(hour\_start, "%02.0f") + ":00:00"  
  
. gen shift\_end\_data\_time = shift\_date + " " + string(hour\_end, "%02.0f") + ":00:00"  
  
. // adjust for across day pattern  
. gen shift\_date\_time\_1 = string(shift\_date\_time + 1, "%td")  
  
.   
. qui replace shift\_end\_data\_time = shift\_date\_time\_1 + " " + string(hour\_end, "%02.0f") ///  
> + ":00:00" if shift\_start == "7 p.m."  
  
.   
.   
. // note, use double to ensure precient  
. gen double shift\_start\_time = clock(shift\_start\_data\_time, "DMY hms")  
  
. gen double shift\_end\_time = clock(shift\_end\_data\_time, "DMY hms")  
  
. format shift\_start\_time %tc  
  
. format shift\_end\_time %tc  
  
.   
. gen double arrive\_time = clock(arrive, "DMY hms")  
  
. gen double leave\_time = clock(leave, "DMY hms")  
  
. format arrive\_time %tc  
  
. format leave\_time %tc  
  
.   
. // calculate percentages  
. // patients arriving before their physician's shift starts  
. gen arrive\_before\_shift = arrive\_time < shift\_start\_time  
  
.   
. // patients discharged after their physician's shift ends  
. gen leave\_after\_shift = leave\_time > shift\_end\_time  
  
.   
. // calculate percentages  
. qui sum arrive\_before\_shift  
  
. qui sum leave\_after\_shift  
  
.

## 2. Data Exploration

I calculated the average predicted severity by half-hour of patient arrival. The connected plot and trend line does not show obvious connection between hours of arrival and the predicted severity of the patient. To test formally test whether patient severity is or is not predicted by hour of the day, I regressed the pred\_lnlos on the dummies of hour arrival variables. The coefficient plot of dummies shows patients stay shorter at dawn and after lunch and stay longer in the morning. However, the result may result from the limit of usage of some inspect equipment such as CT.

. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Question 2\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
. // get hours and minutes  
. gen arrive\_hour = hh(arrive\_time)  
  
. gen arrive\_minute = mm(arrive\_time)  
  
.   
. // compute half-hour  
. gen half\_hour\_interval = arrive\_hour + (arrive\_minute >= 30)/2  
  
.   
. // calculate average servenity by half hours  
. bysort half\_hour\_interval: egen avg\_severity = mean(pred\_lnlos)  
  
.   
. qui twoway (connected avg\_severity half\_hour\_interval,m(o)) ///  
> (lfit avg\_severity half\_hour\_interval, lpattern(dash)), ///  
> ytitle("Average Severity") xtitle("Half-Hour Interval of Day") ///  
> title("Average Severity by Half-Hour of Patient Arrival") ///  
> legend(ring(0) position(4))  
  
. qui graph export "$Out\Average\_Severity.png", replace  
  
.   
. // formally test whether patient severity is or is not predicted by hour of the day  
. qui reg avg\_severity i.arrive\_hour,r  
  
.   
. qui coefplot,keep(\*.arrive\_hour) title("Coefficient of Arrival Hours") baselevels ci vertical ///  
> label xlabel(, angle(45) labsize(vsmall)) yline(0)  
  
. qui graph export "$Out\Coef\_Hours.png", replace

## 3.Estimation and Causal Inference

1. I graphed the census variation relative to end of shift as Fig. 3 shows. The census count – in any census scope, would increase at the beginning of the physician’s shift and decreases as the time get closer to his end of shift time. And the patient under care still decreases even the time passed the shift time for 4 hours.
2. As we have the accurate time of the shift time and the patient arrival time. I construct the lower bound of the census under the criterion that we only take the patients who is under care for the whole hours into account. And I construct the upper bound of the census under the criterion that we counts the patients whoever is under care when he arrives the ED at that hour. At last, the finer census – I excluded the patients whose arrival time is among the last 15 minutes of the hour or leave time is among the first 15 minutes of the hour. One issue one may addressed is that the physician may arrive at the EP before his scheduled time, which would affects the power of our above analysis. In the mean time, the time between the arrival time and leave time of patients may not be accounted in the care time, they may wait at the waiting room or doing some paper works before the physician’s care.
3. If we have the ED data, we may construct the census of the co-work of more than one physicians, and under this circumstance, they may behavior differently.

. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Question 3\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
. // extend shift end by 4 hours  
. gen double shift\_end\_4h\_more = shift\_end\_time + 60\*60\*4000  
  
. format shift\_end\_4h\_more %tc  
  
.   
. save temp.dta, replace  
file temp.dta saved  
  
.   
. use temp.dta, clear  
  
. // creat hourly interval for time shift  
. gen hours\_of\_shift = (shift\_end\_4h\_more - shift\_start\_time) / 3600000  
  
.   
. // create an index for each shift  
. gen shift\_index = \_n   
  
.   
. // expand the dataset for each hour of each shift  
. qui expand hours\_of\_shift  
  
.   
. // generate hour\_id for each hour within the shift  
. bysort shift\_index: gen hour\_id = \_n  
  
.   
. gen double hour\_lb = shift\_start\_time + (hour\_id-1)\*3600000  
  
. gen double hour\_ub = hour\_lb + 3600000  
  
.   
. format hour\_lb %tc  
  
. format hour\_ub %tc  
  
.   
. // lower bound census: counts only throught whole hours  
. gen patient\_under\_care\_lb = (arrive\_time <= hour\_lb) & (leave\_time > hour\_ub)  
  
.   
. bysort phys\_id shift\_date hour\_id: egen census\_lb = sum(patient\_under\_care\_lb)  
  
.   
. // upper bound census: counts if intersects within hours  
. gen patient\_under\_care\_ub = (arrive\_time <= hour\_ub) & (leave\_time > hour\_lb)  
  
.   
. bysort phys\_id shift\_date hour\_id: egen census\_ub = sum(patient\_under\_care\_ub)  
  
.   
. // finer bound census  
. gen patient\_under\_care\_fb = (arrive\_time <= hour\_ub) & (leave\_time > hour\_lb)  
  
. qui replace patient\_under\_care\_fb = 0 ///  
> if (arrive\_time <= hour\_ub) & (leave\_time > hour\_lb) & (leave\_time < hour\_lb + 900000)  
  
. qui replace patient\_under\_care\_fb = 0 ///  
> if (arrive\_time <= hour\_ub) & (arrive\_time > hour\_ub - 900000) & (leave\_time > hour\_lb)  
  
.   
. bysort phys\_id shift\_date hour\_id: egen census\_fb = sum(patient\_under\_care\_fb)  
  
.   
. // end of shift  
. gen end\_of\_shift = hour\_id - hours\_of\_shift + 4  
  
. save patients\_all.dta, replace  
file patients\_all.dta saved  
  
.   
. use patients\_all.dta, clear  
  
. qui duplicates drop phys\_id shift\_date shift\_start shift\_end end\_of\_shift, force  
  
. keep phys\_id shift\_date shift\_start shift\_end hour\_id ///  
> patient\_under\_care\_lb patient\_under\_care\_ub patient\_under\_care\_fb end\_of\_shift  
  
. rename hour\_id hour  
  
. save census.dta, replace  
file census.dta saved  
  
.   
. use census.dta, clear  
  
. // How does the census vary with time relative to end of shift  
. bysort end\_of\_shift: egen sum\_patient\_under\_care\_fb = sum(patient\_under\_care\_fb)  
  
. bysort end\_of\_shift: egen sum\_patient\_under\_care\_lb = sum(patient\_under\_care\_lb)  
  
. bysort end\_of\_shift: egen sum\_patient\_under\_care\_ub = sum(patient\_under\_care\_ub)  
  
.   
. qui duplicates drop end\_of\_shift, force  
  
. qui twoway (connected sum\_patient\_under\_care\_fb end\_of\_shift) ///  
> (connected sum\_patient\_under\_care\_lb end\_of\_shift) ///  
> (connected sum\_patient\_under\_care\_ub end\_of\_shift), xline(0, lpattern(dash)) ///  
> ytitle("Census Count") xtitle("Time Relative to End of Shift (Hours)") ///  
> title("Census Variation Relative to End of Shift") xtick(-9(1)4) ///  
> legend(ring(0) pos(11))  
  
. qui graph export "$Out\Census\_Variation.png", replace  
  
.

## 4. Further Analysis

I regressed the length of the stay on the dummies of the physicians, the result shows at the Figure 4, phys\_id=16 is the one who is fastest at discharging patients. The potential threats may be that different physicians in different ED may encounter different types of patients, thus leading to different discharge time. So we can control the expected log length of stay, where length of stay is the difference between leave and arrive, based on patient demographics and medical conditions. The result is robust, physician 16 and 30 are two who are fastest at discharging patients.

Moreover, a potential issue may arising from the fact that the patient may be discharged after the physician’s shift time. So we just regress the log length of stay on the dummies of time to shift. The result is still robust to this specification: physician 16 and 30 are two who are fastest at discharging patients.

. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Question 4\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
. use temp.dta, clear  
  
.   
. // gen length of stay(seconds)  
. gen double log\_length\_stay = log((leave\_time - arrive\_time)/1000)  
(4 missing values generated)  
  
.   
. qui reg log\_length\_stay i.phys\_id, r  
  
. qui coefplot,keep(\*.phys\_id) title("Coefficient of Physician") baselevels ci vertical ///  
> label xlabel(, angle(45) labsize(vsmall)) yline(0)  
  
. qui graph export "$Out\Coef\_phys.png", replace  
  
.   
. // control pred\_lnlos  
. qui reg log\_length\_stay i.phys\_id pred\_lnlos, r  
  
. qui coefplot,keep(\*.phys\_id) title("Coefficient of Physician, Controlled for pred\_los") baselevels ci vertical ///  
> label xlabel(, angle(45) labsize(vsmall)) yline(0)  
  
. qui graph export "$Out\Coef\_phys\_control.png", replace  
  
.   
.   
. // los versus time to shift  
. use patients\_all.dta, clear  
  
. gen double log\_length\_stay = log((leave\_time - arrive\_time)/1000)  
(52 missing values generated)  
  
.   
. qui reg log\_length\_stay i.phys\_id##i.hour\_id, r  
  
. qui coefplot,keep(\*.phys\_id) title("Coefficient of Physician, Controlled for time to shift") baselevels ci vertical ///   
> label xlabel(, angle(45) labsize(vsmall)) yline(0)  
  
.   
. qui graph export "$Out\Interaction\_coef.png", replace  
  
.

1. I finished this markdown file by markstat. The source code is in my github repository, [here](https://github.com/whuhu/) [↑](#footnote-ref-20)