

**Replication Package for “Competition and Product Composition: Evidence from Hollywood”
(*International Journal of Industrial Organization*, forthcoming)**

Joseph Kuehn

Ryan Lampe

June 26, 2023

1) Overview

This replication package contains the data and code to generate Tables 1-8 and Figures 1-3 in “Competition and Product Composition: Evidence from Hollywood.”

2) Dataset

The main dataset is `moviedata_wmonth.dta`. The dataset contains revenues and movie details for wide release movies between 2009-2018. The replication package also contains the dataset `deflator.dta` which contains the yearly price index used to deflate revenues.

3) Computational Requirements

The replication package uses both Stata (Version 14.0 was used) and Matlab (version R2022a was used). In total, the entire replication took approximately 2 months on a Mac Studio with 128 GB of RAM. Most of the computational time was in simulating the counterfactuals.

4) Steps to Replicate Results

a) Run `sumstats.do`

- Uses data in `moviedata_wmonth.dta` and `deflator.dta`
- Generates the summary statistics in **Table 1**
- Generates the figures in **Figure 1**
- Generates the regression results in **Table 2**
- Generates the figures in **Figure A2**

b) Run `sumfigure.do` and `sumfigure.m`

- Generates **Figure 2**

c) Run `logit.do`

- Uses data in `moviedata_wmonth.dta` and `deflator.dta`
- Runs the nested logit regression in **Table 3**
- Extracts the movie fixed effects from the regression results (the thetas)
- Exports the data to excel file, `moviedata_form.xlsx`, to use for structural estimation
- At the movie level, regresses the fixed effects on budget, distributor fixed effects, and budget tier fixed effects to get results in **Table 4**

- Gets the distribution of budgets for each budget tier and saves to distribution.xlsx to use later in the estimation (in **Table 5**) of paper

d) Run adjustreleaseweek_sim.m

- Uses data in moviedata_form.xlsx, logitresults.xlsx, distribution.xlsx, and thetaregs.xlsx (all created in logit.do)
- Other code used: findms_nest.m (calculates market shares based on data and parameters), expectedrevenue.m (calculates expected revenue of film based on shocks, data, and parameters), and weekadjust.m (adjusts the release date timing game for a given year)
- Determines the expected quality of each film
- Based on expected quality, determines the release date calendar for each year assuming that all shocks are initially set to zero
- Draws the movie-week demand shocks, the movie quality shocks, and the movie release date shocks. Will adjust release dates to these shocks in fixedcostestimation_sim.m

e) Run fixedcostestimation_sim.m

- Other code used: addmovie.m (adds movie) and dropmovie.m (drops movie)
- First allows studios to adjust their release dates in response to shocks
- Estimates the fixed cost bounds in **Table 6**

f) Run cf_nothing_sim.m

- Allows studios to add or drop movies and adjust their release dates
- Provides the base case where nothing changes to compare the counterfactuals to

g) Run the counterfactuals in cf_main_startbench_sim.m, cf_main_alt_startbench_sim.m, cf_merger_startbench_sim.m, cf_merger_alt_starbench_sim.m, cf_dropstudio_startbench_sim.m, cf_addstudio_startbench_sim.m

- Each simulates one of the 6 counterfactuals in **Table 7**
- cf_main_startbench_sim.m: Sets Netflix subscribers to zero
- cf_main_alt_startbench_sim.m: Sets Netflix subscribers to pre-2011 numbers
- cf_merger_startbench_sim.m: Mergers Disney and Fox (where Fox movies obtain Disney fixed effect in quality)
- cf_merger_alt_startbench_sim.m: Mergers Disney and Fox (where Fox movies do not obtain Disney fixed effects in quality)
- cf_dropstudio_startbench_sim.m: Studio exits
- cf_addstudio_startbench_sim.m: New studio enters

h) Run welfare.m

- Cycles through each error draw, year, and studio, and calculates the own change in revenue and change to other studios revenues from dropping a movie in each tier and adding a movie in each tier
- Results are in **Table 8**

- i) Run `explaineffect_netflix11.m`
- Produces **Figure 3**
 - For each budget tier, one by one drops the observed movies in that tier and one-by-one adds additional movies in that tier, to get the total change in profit
- j) Run `explaineffect_merger.m`
- Similar to `explaineffect_netflix11.m`, but only looks at Fox and comparison is to counterfactual where Disney acquires Fox
 - Use to produce **Figure A5** and **Figure A6**
- k) Run `explaineffect_lorenz.m`
- Use to produce **Figure A3**

Tables		Figures	
Table 1	<code>sumstats.do</code>	Figure 1	<code>sumstats.do</code>
Table 2	<code>sumstats.do</code>	Figure 2	<code>Sumfigure.m</code>
Table 3	<code>logit.do</code>	Figure 3	<code>explaineffect_netflix.m</code>
Table 4	<code>logit.do</code>		
Table 5	<code>logit.do</code>		
Table 6	<code>fixedcostestimation_sim.m</code>		
Table 7	<code>cf_main_startbench_sim.m</code> , <code>cf_main_alt_startbench_sim.m</code> , <code>cf_merger_startbench_sim.m</code> , <code>cf_merger_alt_starbench_sim.m</code> , <code>cf_dropstudio_startbench_sim.m</code> , <code>cf_addstudio_startbench_sim.m</code>		
Table 8	<code>welfare.m</code>		