Project in applied econometrics
Report

Lucas Javaudin, Robin Le Huérou-Kérisel, Rémi Moreau

March 2018

Abstract

This project has aimed at reproducing Moretti's 2011 paper on social learning effects in movie sales with R. We also blabla. Main results:

Contents

1 Intuitions and detailed presentation of the model

1.1 Some intuitions

1.2 Presentation of the model

bonjour je m'appelle Rémi

2 Analysis and main results

Moretti's purpose is to provide evidence of social learning in consumption, that is to say that people tend to take into account their peers' experience to get a more precise idea of the value of a good. Economists, Moretti says, have had difficulties showing such social learning effects because of the absence of useful microdata on the matter. Moretti's innovation lies in his use of market-level data to identify social learning. He does so by defining what he calls "surprises" in movie sales: surprises, as their name suggests, consist in the difference between expected and actual sales. Moretti proposes that if we observe a surprise, we should also observe social learning effects: if a film is better or worse than expected, then by gathering experience through peers, people should reconsider their expectations and we might be able to see it in the data. In particular, Moretti makes five predictions on things we should be able to observe in presence of social learning:

1. in presence of social learning, sales of movies with positive and negative surprises should diverge: sales of better-than-expected movies should decrease at a lower rate than worse ones (see 2.2);
2. we should observe less social learning effects from a movie on which quality we have a precise idea and more social learning effects from movies which have a more uncertain quality (see 2.3);
3. we should observe more social learning effects when people have a greater social network (see 2.4);
4. we should be able to observe that the effects of a surprise decline over time: once the information on the quality of a movie has been shared, what was a surprise should not play a major role in sales (see 2.5);
5. we should not observe social learning effects when a surprise is due to elements other than quality of the film (let say weather).

We have replicated Moretti's work and tried to confront his predictions with French data.

2.1 Identification of the surprises

Surprises consist in the residuals of the regression of the log-number of sales in the first week on the log-number of screens available (opened by theaters). This definition of surprises holds because we suppose that theaters are profit-maximizing agents and make use of all the available information to predict the success of a movie. If this definition is correct, we should expect log-number of screens opened by theaters first week to be a good indicator of knowledge available on the movie quality before it is released. In the Table 1 we reproduce Moretti's regression of *log_sales_first_we* on *log_screens_first_week*. Each column is the result of the regression when we control with some variables (film genre, rating available, cost, distributor, weekday, month, week, year). The fact that adding control variables doesn't change the robustness of the regression proves Moretti's point which is that theaters take into account these factors when deciding their number of available screens.

Table 1: Regression of first-weekend sales on number of screens

	Dependent variable:						
	log_sales_first_we						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
log_screens_first_week	0.893***	0.896***	0.883***	0.871***	0.803***	0.806***	0.813***
	(0.004)	(0.005)	(0.005)	(0.005)	(0.006)	(0.006)	(0.006)
R^2	0.907	0.909	0.910	0.912	0.932	0.936	0.938
Adjusted R^2	0.907	0.908	0.910	0.912	0.928	0.931	0.933
Note:							*p<0.1; **p<0.05; ***p<0.01

We have performed the same kind of regression on France data from 2004 to 2008 and find quite similar results (see table 2).

Table 2: Regression of first-week entries on number of screens for France

	Dependent variable:						
	log_entree_fr						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
log_seance_fr	1.208***	1.237***	1.237***	1.279***	1.282***	1.287***	1.196***
	(0.009)	(0.010)	(0.010)	(0.014)	(0.014)	(0.014)	(0.014)
R^2	0.893	0.899	0.900	0.917	0.924	0.925	0.943
Adjusted R^2	0.893	0.898	0.898	0.910	0.915	0.916	0.935
Note:							*p<0.1; **p<0.05; ***p<0.01

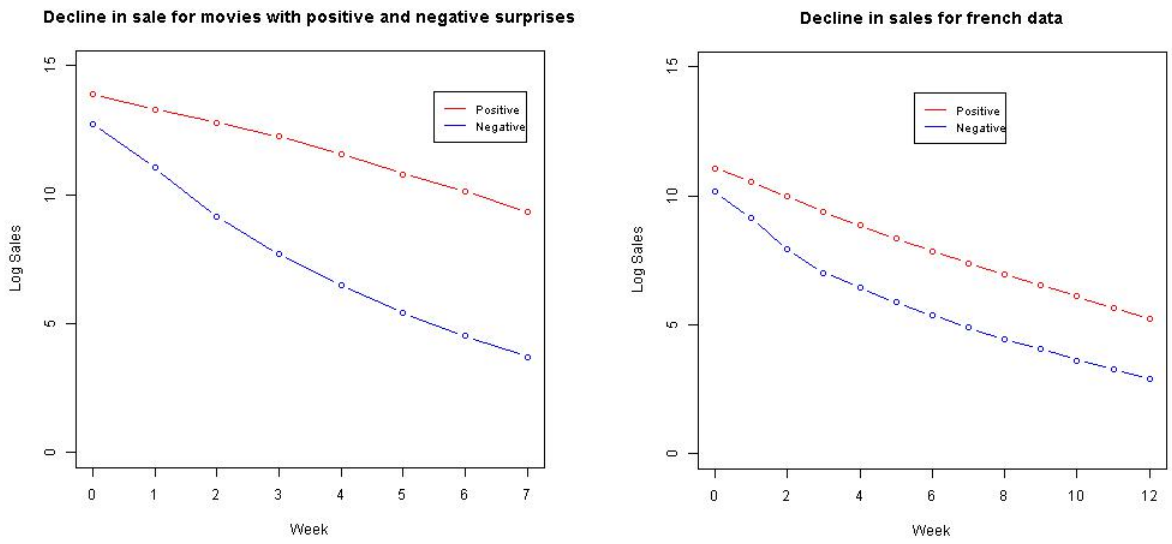
We can see that the number of sales in first week is highly explained by the number of screens opened. This result holds even when adding controls: each column corresponds to a regression in which we added a control variable (genre, ratings, distributors, month and week, year, and some other variables).

Figure 1: R code used to obtain French surprises

```
# Regression of first week sales on number of screens.
regSurprise1 <- lm(log_entree_fr ~ log_seance_fr, data = df, subset = (t==0))
# Including dummies for genre
regSurprise2 <- lm(log_entree_fr ~ log_seance_fr + genre, data = df, subset = (t==0))
# Including dummies for ratings
regSurprise3 <- lm(log_entree_fr ~ log_seance_fr + genre + interdiction, data = df, subset = (t==0))
# Including dummies for distributor
regSurprise4 <- lm(log_entree_fr ~ log_seance_fr + genre + interdiction + id_distributeur, data = df,
    subset = (t==0))
# Including dummies for month and week
regSurprise5 <- lm(log_entree_fr ~ log_seance_fr + genre + interdiction + id_distributeur + factor(mois
    ) + factor(semaine), data = df, subset = (t==0))
# Including dummies for year
regSurprise6 <- lm(log_entree_fr ~ log_seance_fr + genre + interdiction + id_distributeur + factor(mois
    ) + factor(semaine) + factor(annee), data = df, subset = (t==0))
# Including other variables
regSurprise7 <- lm(log_entree_fr ~ log_seance_fr + genre + interdiction + id_distributeur + factor(mois
    ) + factor(semaine) + factor(annee) + MoyennePresse + MoyenneSpectateur + PoidsCasting + pub, data
     = df, subset = (t==0))
```

2.2 Divergence of the sales

Figure 2: Comparing decline in sales between Moretti's and French data



The first prediction of Moretti is that if there are social learning effects in movie sales, we should observe diverging trajectories between movies with positive and negative surprises. The idea is simple: without social learning, sales of movies with positive and negative surprises should decrease at the same rate; in other words, surprises would not have any effect on sales. Indeed, people would not take surprises as a new information on the movie quality.

Moretti estimates models of the form:

$$\ln(y_{jt}) = \beta_0 + \beta_1 * t + \beta_2(t * S_j) + d_j + u_{jt} \tag{1}$$

where $\ln(y_{jt})$ is the log of box-office sales in week t; S_j is surprise; d_j is a movie fixed effect. The variable of interest is β_2 because we want to identify an effect of the surprise on the dynamic of sales over time.

In the figure 2, we have reproduced Moretti's graph and plotted the graph for French data. In Moretti's graph we clearly see the diverging trajectories of the sales. Our graph also shows diverging trajectories

Table 3: Decline in box-office sales by opening week surprise for French data

	Dependent variable:			
	log_entree_fr			
	(1)	(2)	(3)	(4)
t	-0.526^{***}	-0.526^{***}	-0.571^{***}	
	(0.002)	(0.002)	(0.003)	
t:surprise		0.076^{***}		
		(0.004)		
t:positive_surprise			0.087^{***}	
			(0.004)	
t:bottom_surpriseFALSE				-0.459^{***}
				(0.004)
t:bottom_surprise				-0.574^{***}
				(0.004)
t:middle_surprise				-0.088^{***}
				(0.005)
Observations	26,598	26,598	26,598	26,598
R^2	0.851	0.853	0.853	0.854
Adjusted R^2	0.838	0.841	0.841	0.841
Note:				*p<0.1; **p<0.05; ***p<0.01

En fait je crois que je comprends pas tout ici.

Figure 3: R code used to obtain French sales dynamics

```
##############################################
#  Prediction 1: Surprises and Sale Dynamics  #
##############################################

# In this part, we study the difference in rate of decline between movies with a positive surprise and
    movies with a negative surprise.

# Regression of sales on the interaction between time and surprises.
# We use the command felm of the package lfe to compute linear regressions with thousands of dummies.
regSaleDynamics1 <- felm(log_entree_fr ~ t | X, data = df)
regSaleDynamics2 <- felm(log_entree_fr ~ t + t : surprise | X, data = df)
regSaleDynamics3 <- felm(log_entree_fr ~ t + t : positive_surprise | X, data = df)
regSaleDynamics4 <- felm(log_entree_fr ~ t : bottom_surprise + t : middle_surprise | X, data = df)

# Print a table with the results of the regressions.
stargazer(regSaleDynamics1, regSaleDynamics2, regSaleDynamics3, regSaleDynamics4, omit.stat=c("f", "ser
    "), title='Decline in box-office sales by opening week surprise')
```

2.3 Precision of the prior

Another prediction of Moretti is that the effect of surprises should vary with the precision of the prior people have on movies.

Moretti estimates models of the form:

$$\ln(y_{jt}) = \beta_0 + \beta_1 * t + \beta_2(t * S_j) + \beta_3(t * precision_j) + \beta_4(t * S_j * precision_j) + d_j + u_{jt} \quad (2)$$

Table 4: Precision of the prior

	Dependent variable:		
	log_entree_fr		
	(1)	(2)	(3)
t	-0.570^{***}	-0.698^{***}	-0.678^{***}
	(0.003)	(0.013)	(0.004)
t:positive_surprise	0.105^{***}	0.109^{***}	0.009
	(0.005)	(0.018)	(0.006)
t:saga	-0.027		
	(0.016)		
t:positive_surpriseTRUE:saga	-0.145^{***}		
	(0.019)		
t:var_surprise		0.370^{***}	
		(0.035)	
t:positive_surpriseTRUE:var_surprise		-0.062	
		(0.050)	
t:art_essai			0.259^{***}
			(0.006)
t:positive_surpriseTRUE:art_essai			0.066^{***}
			(0.008)
Observations	26,598	26,546	26,598
R^2	0.855	0.854	0.880
Adjusted R^2	0.843	0.842	0.870
Note:			*p<0.1; **p<0.05; ***p<0.01

2.4 Size of the Social Network

Consumers with a larger social network receive more feedbacks from their peers and thus they are able to evaluate more precisely the quality of the movie.

Table 5: Precision of peers' signal

	Dependent variable:	
	log_entree_fr	
	(1)	(2)
t	-0.663^{***}	-0.451^{***}
	(0.007)	(0.005)
t×positive_surprise	0.061^{***}	0.076^{***}
	(0.010)	(0.006)
t×tout_public	0.115^{***}	
	(0.008)	
t×positive_surprise×tout_public	0.031^{***}	
	(0.011)	
t×seance_fr_first_week		-0.033^{***}
		(0.001)
t×positive_surprise×seance_fr_first_week		0.011^{***}
		(0.001)
Observations	26,598	26,598
R^2	0.856	0.867
Adjusted R^2	0.844	0.856
Note:	*p<0.1; **p<0.05; ***p<0.01	

2.5 Does learning decline over time?

Table 6: Convexity of the sales profile

	Dependent variable:
	log_entree_fr
t	-0.978^{***}
	(0.011)
t^2	0.034^{***}
	(0.001)
$t \times$positive_surprise	0.393^{***}
	(0.016)
$t^2 \times$positive_surprise	-0.026^{***}
	(0.001)
Observations	26,598
R^2	0.861
Adjusted R^2	0.850
Note:	*p<0.1; **p<0.05; ***p<0.01

3 Conclusion: some comments

A R codes

Data cleaning

```
####################
#  Data Cleaning  #
####################

# In this part, we change the dataset to make it closer to the dataset of Moretti.

# Remove the movies without any screen in France during the first week (667 movies).
fr_df <- fr_df[!is.na(fr_df$seance_fr1),]
# Remove the movies without any id_distributeur (4 movies).
fr_df <- fr_df[!is.na(fr_df$id_distributeur),]

# Set MoyennePresse and MoyenneSpectateur to the mean if no value is specified.
mean_moy <- mean(fr_df[!is.na(fr_df$MoyennePresse), 'MoyennePresse'])
fr_df[is.na(fr_df$MoyennePresse), 'MoyennePresse'] <- mean_moy
mean_moy <- mean(fr_df[!is.na(fr_df$MoyenneSpectateur), 'MoyenneSpectateur'])
fr_df[is.na(fr_df$MoyenneSpectateur), 'MoyenneSpectateur'] <- mean_moy

# Repeat each columns 13 times.
n <- nrow(fr_df)
df <- fr_df[rep(1:n, each=13),]

# Add a column to indicate the week.
df$t <- rep(0:12, n)

# Replace the variables for each week (e.g. 'entree_paris1') with a global variable (e.g. 'entree_paris')
for (i in 0:12) {
for (variable in c('entree_paris', 'seance_paris', 'entree_fr', 'seance_fr')) {
# Concatenate the variable name with and indicator for the week (e.g. 'entree_paris1').
variable_t <- paste(c(variable, toString(i+1)), collapse='')
# For each week, the variable in the new df (e.g. 'entree_paris') is taken from the old df (e.g. 'entree_paris1').
df[df$t==i, variable] <- fr_df[,variable_t]
}
}

# Keep only the useful variables.
df <- df[,c(1:6, 33:43, 70:85)]

# Replace the NAs in seance_fr with zeros.
df[is.na(df$seance_fr), 'seance_fr'] <- 0

# Generate logarithm of sales and screens.
df$log_entree_paris <- log(df$entree_paris + 1)
df$log_seance_paris <- log(df$seance_paris + 1)
df$log_entree_fr <- log(df$entree_fr + 1)
df$log_seance_fr <- log(df$seance_fr + 1)

# Variable id_distributeur is a factor.
df$id_distributeur <- as.factor(df$id_distributeur)

# Variable id is a factor (this is used for movie dummies with the package lfe).
df$X <- as.factor(df$X)
df$X.eff <- rnorm(nlevels(df$X))
```