Online appendix for Rational Illiquidity and Consumption: Theory and Evidence from Income Tax Withholding and Refunds

Michael Gelman, Shachar Kariv, Matthew D. Shapiro, and Dan Silverman*

*Gelman: Claremont McKenna College (mgelman@cmc.edu); Kariv: University of California, Berkeley (kariv@berkeley.edu); Shapiro: University of Michigan and NBER (shapiro@umich.edu); Silverman: Arizona State University and NBER (dsilver3@asu.edu).

A Appendix – Data Filters, Definitions

The main analysis sample is filtered appropriately to reduce measurement error in key variables and to focus attention on workers with at least some regular paycheck income. In particular, to observe a sufficiently complete view of spending and income, we limit attention to app users who link all (or most) of their accounts to the app, generate a long time series of observations, and have positive income in each month. To study the importance of paycheck vs non-paycheck income, we also restrict attention to app users who receive regular bi-weekly paychecks throughout most of the time we observe them in our data. The consequences for sample size are presented in Table A.1 below.

A.1 Defining Account Linkage

The analysis may be biased if all accounts that are used for receiving income and making expenditures are not observed. For example, an individual may have a checking account that is used to pay most bills and a credit card that it used when income is low. If credit card expenditures are not properly observed the MPC will be biased downwards.

In order to identify linked accounts, we use a method that calculates how many credit card balance payments are also observed in a checking account. We define the variable *linked* as the ratio of the number of credit card balance payments observed in all checking accounts that matches a particular payment that originated from all credit card accounts. For example, a typical individual will pay their credit card bill once a month. If they existed in the data for the whole year, they will have 12 credit card balance payments. If 10 of those credit card payments can be linked to a checking account the variable $linked = \frac{10}{12} \approx 0.83$.

One drawback to this approach is that it requires individuals to have a credit card account. To ensure that those without credit cards are still likely to have linked accounts, we also condition on individuals who have three or more accounts.

A.2 Defining Regular Paycheck

In order to identify regular paychecks, we start by using keywords that are commonly associated with these transactions.[1] We condition on four statistics to ensure that these transactions represent regular paychecks.

1. Number of paychecks ≥ 5
2. Median paycheck amount $>$ \$200
3. Median absolute deviation of days between paychecks is ≤ 5
4. Coefficient of variation of the paycheck amount ≤ 1

A.3 Defining Stable Paycheck

The ratio of paycheck and non-paycheck income is an essential ingredient in our model. To ensure we are estimating the ratio correctly, we restrict attention to users who have received a paycheck at least 2/3 of the time we observe them in the sample.

A.4 Payroll Periodicity

We limit the sample to individuals with bi-weekly payroll. Bi-weekly paychecks are identified as a series of paychecks with the median number of days between each paycheck equalling 14 days.

A.5 Sample Size

Table A.1 shows the evolution of the sample size from all users in the sample to those that survive the selection criteria. The criteria selects users who have a long time series ($\geq$ 40 months), a high linked account ratio ($\geq$ 0.8), a reasonable number of accounts linked ([3,15]), and receive a regular bi-weekly paycheck. We choose to drop users that have over 15 accounts linked because these accounts typically represent business users. Table 2 shows

[1] Keywords used to identify paychecks are "dir dep","dirde p","salary","treas xxx fed","fed sal","payroll","ayroll","payrll","payrl","payrol","pr payment","adp","dfas-cleveland","dfas-in" and DON'T include the keywords "ing direct","refund","direct deposit advance","dir dep adv."

that this final sample compares well with external data for the variables that are important in our analysis.

Table A.1: Effect of sample filters

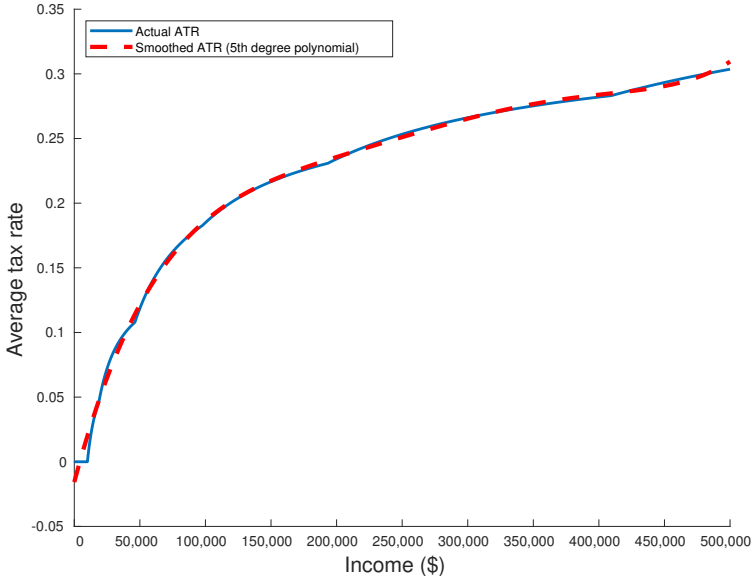
	Individuals	%
Full sample as of December 2012	883,529	100
Long time series (N $\geq$ 40)	321,092	36
Linked ratio $\geq$ 0.8	244,282	28
Linked accounts $\in$ [3,15]	192,408	22
Has regular bi-weekly paycheck	92,883	11
Has stable paycheck	62,946	7

A.6 2013 Tax Schedule

The tax function is based on 2013 average tax rates (ATR). It is calculated from the Stata package *taxliab* for income values over the range \$0 to \$500,000 in \$100 intervals. The package calculates the ATR from the marginal tax rate schedule. We assume that individuals are single filers who claim two personal exemptions (\$3,900 each) and the standard deduction (\$6,100).[2] We then approximate the ATR schedule with a 5th degree polynomial. The actual and smoothed schedule is shown in Figure A.1. Note that while the smoothed function is negative for very low levels of income, income in the model is never this low.

[2]These values are taken from IRS publication 501 (https://www.irs.gov/pub/irs-prior/p501–2013.pdf).

Figure A.1: Actual and smoothed average tax rate function



Notes: This table plots the actual and smoothed average tax rate function. The smoothed average tax rate are calculated using a 5th degree polynomial.

The tax liability function is then defined as

$$\tau(Y) = ATR(Y) \times Y$$

where Y is income and $ATR(\cdot)$ represents the smoothed average tax rate function plotted above.

B Appendix – Estimating Gross Paycheck Income

In our model, an individual makes withholding and saving decisions based on gross (pre-withheld) paycheck income and non-withheld income. In our data, we only observe net (post-withheld) income so we estimate gross paycheck income based on which taxes are withheld from an individuals' paycheck income.

The various types of withholding are

1. Federal income tax withholding (based on the yearly withholding schedule published by the IRS under Publication 15 or "Circular E")

2. Social security payroll tax (6.2%)

3. Medicare tax (1.45%)

4. State and local tax (based on yearly average state and local taxes collected)[3]

The observed net paycheck income is a function of gross paycheck income

$$\tilde{p}_{i,b,t} = f(p_{i,b,t}; s, e, t) \tag{1}$$

where s represents filing status, e represents the number of exemptions, and t represents year. We assume single filing status with two exemptions. We then invert this function to recover gross paycheck income.

Pre-tax benefits such as health insurance premiums and 401(k) contributions also lead to differences in gross and net paycheck income. We do not adjust for these benefits as we do the types of withholding listed above. We don't see this income, but equally we dont see its consumption. Moreover, these benefits are generally not subject to income taxation that we are modeling. Hence, that they are excluded from both income and spending in the data is fortuitously correct. The same argument holds for pension benefits, but with a more complicated intertemporal accounting.

C Appendix – The Withholding Function

C.1 Measuring Excess Withholding Due to High Frequency Paycheck Volatility

As noted in section 3, the rules governing paycheck tax withholding schedules, and the convexity of the tax schedule, may induce a "mechanical" relationship between within-year paycheck volatility and refunds. To quantify the magnitude of the mechanical effect we define excess withholding from high frequency paycheck volatility as:

[3]We take total state and local income tax collected from "U.S. Census Bureau, Quarterly Summary of State and Local Government Tax Revenue" and divide it by total payroll tax reported in "IRS, Statistics of Income Division, Publication 1304" to arrive at an average state and local tax rate. The rates are 5.320%, 5.154%, 4.921%, and 5.291% for 2013, 2014, 2015, and 2016 respectively.

$$ExcessW_{i,y} = \sum_{b=1}^{26} (w(p_{i,b,t}; s, e, t) - w(\bar{p}_{i,t}; s, e, t)) \tag{2}$$

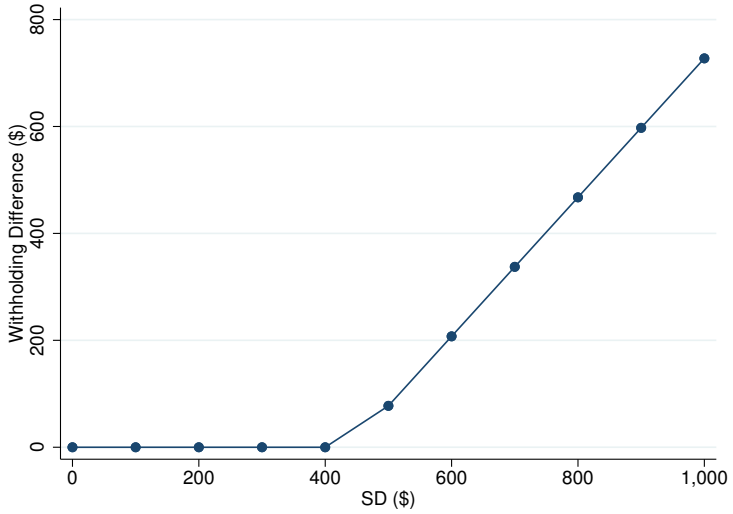
where $w(\cdot; s, e, t)$ is a periodic withholding function that takes paycheck income as its argument and is influenced by filing status s, number of exemptions e, and year t.[4] $p_{i,b,t}$ is the bi-weekly pre-withholding paycheck for individual i in bi-week b of year t, and $\bar{p}_{i,t}$ is the average bi-weekly pre-withholding paycheck for individual i in year t.[5] We assume single filing status and two exemptions in our calculations of excess withholding.

Figure C.1 illustrates the relationship between this measure of potential excess withholding and within-year paycheck volatility. The example in the figure assumes paychecks are one standard deviation above average half the time and one standard deviation below average the other half of the time. As expected, the measure of potential excess withholding, $ExcessW_{i,t}$, increases as within year paycheck variation increases. The relationship is not linear, however, because potential excess withholding is positive only if annualized paycheck income crosses marginal tax rates. Because the tax schedule is a piece-wise linear function of income, there are regions where modest within-year variation doesn't lead to any excess withholding.

[4]The withholding function is based on the actual withholding schedule in form IRS publication 15 (aka circular E) https://www.irs.gov/pub/irs-pdf/p15.pdf.

[5]We do not observe pre-withholding income $p_{i,b,t}$ directly. Instead we observe post-withholding income $\tilde{p}_{i,b,t} = p_{i,b,t} - w(p_{i,b,t}; s, e, y)$. Therefore, we estimate $p_{i,b,t}$ from $\tilde{p}_{i,b,t}$ conditional on s, e, and t. Because observed post-withholding paycheck income is a function of pre-withholding income and other tax parameters, $\tilde{p}_{i,b,t} = f(p_{i,b,t}; s, e, t)$ and we can simply take the inverse of this function to estimate $p_{i,b,t}$ by $p^*_{i,b,t} = f^{-1}(\tilde{p}_{i,b,t}; s, e, t)$.

Figure C.1: $ExcessW_y$ as a function of within-year paycheck variation



Notes: $ExcessW_{i,y}$ is calculated based on a single filer with two exemptions. The paycheck fluctuates one standard deviation above the average half of time and one standard deviation below the average the rest of time.

C.2 Calibrating the Withholding Function

For purposes of calculating excess withholding, the withholding function is calibrated using IRS publication 15 (aka circular E).[6] Figure C.2 displays an example of a table used to calibrate the withholding for individuals who receive a bi-weekly paycheck. We calibrate a withholding function for each year to account for the yearly changes in the schedules.

Figure C.2: Withholding table example

TABLE 2—BIWEEKLY Payroll Period

(a) SINGLE person (including head of household)—				**(b) MARRIED person**—			
If the amount of wages (after subtracting withholding allowances) is:		The amount of income tax to withhold is:		If the amount of wages (after subtracting withholding allowances) is:		The amount of income tax to withhold is:	
Not over $88		$0		Not over $333		$0	
Over—	**But not over—**		**of excess over—**	**Over—**	**But not over—**		**of excess over—**
$88	—$447	$0.00 plus 10%	—$88	$333	—$1,050	$0.00 plus 10%	—$333
$447	—$1,548	$35.90 plus 15%	—$447	$1,050	—$3,252	$71.70 plus 15%	—$1,050
$1,548	—$3,623	$201.05 plus 25%	—$1,548	$3,252	—$6,221	$402.00 plus 25%	—$3,252
$3,623	—$7,460	$719.80 plus 28%	—$3,623	$6,221	—$9,308	$1,144.25 plus 28%	—$6,221
$7,460	—$16,115	$1,794.16 plus 33%	—$7,460	$9,308	—$16,360	$2,008.61 plus 33%	—$9,308
$16,115	—$16,181	$4,650.31 plus 35%	—$16,115	$16,360	—$18,437	$4,335.77 plus 35%	—$16,360
$16,181		$4,673.41 plus 39.6%	—$16,181	$18,437		$5,062.72 plus 39.6%	—$18,437

Source: IRS publication 15 (aka circular E) https://www.irs.gov/pub/irs-pdf/p15.pdf.

[6] https://www.irs.gov/pub/irs-pdf/p15.pdf

D Appendix – Predicted Refunds Statistics

This appendix shows summary statistics for the regression reported in Table 9, column (2).

Table D.1: Summary statistics for each predicted refund quintile ($)

Q_i^j	Mean	p25	p50	p75
1	2,611	2,546	2,623	2,686
2	2,859	2,803	2,859	2,913
3	3,095	3,031	3,092	3,157
4	3,395	3,305	3,389	3,482
5	3,986	3,732	3,908	4,174
Total	3,189	2,803	3,092	3,482

E Appendix – Correlates of Tax Refunds by Terciles of Income

Table E.1: Tax refunds and income volatility by income tercile: $Log(Refund)_{it}$

	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Panel A: Top							
$payshare_i$	-1.182				-0.903	-1.170	
	(0.0396)				(0.0588)	(0.0456)	
$Log(\sigma^2_{\nu_i^{NP}})$		0.0856	0.0902		0.0424		0.0928
		(0.00462)	(0.00449)		(0.00665)		(0.00550)
$Log(\sigma^2_{\nu_i^{P}})$			-0.0220				
			(0.00428)				
$Log(ExcessW_{it-1})$				0.0143	0.0123		
				(0.00259)	(0.00256)		
NxT	46,380	46,380	46,380	29,405	29,405	29,405	29,405
N	26,418	26,418	26,418	21,391	21,391	21,391	21,391
R^2	0.051	0.038	0.039	0.015	0.051	0.047	0.038
Panel B: Middle							
$payshare_i$	-0.829				-0.945	-0.881	
	(0.0350)				(0.0506)	(0.0417)	
$Log(\sigma^2_{\nu_i^{NP}})$		0.0419	0.0433		-0.00708		0.0476
		(0.00353)	(0.00355)		(0.00503)		(0.00418)
$Log(\sigma^2_{\nu_i^{P}})$			-0.0165				
			(0.00394)				
$Log(ExcessW_{it-1})$				0.00997	0.0173		
				(0.00286)	(0.00287)		
NxT	48,566	48,566	48,566	31,061	31,061	31,061	31,061
N	29,335	29,335	29,335	23,250	23,250	23,250	23,250
R^2	0.029	0.016	0.017	0.015	0.035	0.034	0.021
Panel C: Bottom							
$payshare_i$	-0.295				-0.132	-0.341	
	(0.0374)				(0.0542)	(0.0423)	
$Log(\sigma^2_{\nu_i^{NP}})$		0.0469	0.0422		0.0423		0.0524
		(0.00375)	(0.00375)		(0.00541)		(0.00426)
$Log(\sigma^2_{\nu_i^{P}})$			0.0419				
			(0.00432)				
$Log(ExcessW_{it-1})$				0.0377	0.0374		
				(0.00388)	(0.00392)		
NxT	39,806	39,806	39,806	27,246	27,246	27,246	27,246
N	21,142	21,142	21,142	18,025	18,025	18,025	18,025
R^2	0.006	0.010	0.014	0.008	0.016	0.007	0.012

Notes: Dependent variable is $Log(Refund)_{it}$. Robust standard errors in parenthesis. NxT represents the number of individual-year observations. N represents the number of individual observations. Columns (4) and (5) are based on one fewer year's observations to allow for the lagged variable. Columns (6) and (7) repeat the estimates of columns (1) and (2) with this sample.

F Appendix – Solution Method

We use a combination of traditional value function iteration and the endogenous grid method to solve the maximization problem in three steps.

1. Step 1: Solve for optimal S and $\hat{W}$ when both are positive
   (a) Assume a grid of values for the control variable S_t
   (b) Conditional on S_t, use the FOC for $\widehat{W}_t$ to solve for $\widehat{W}_t$. $u'(C_t(\widehat{W}_t)) = \beta \int_\nu u'(C_{t+1}(\widehat{W}_t))\tilde{\Phi} d\nu$
   (c) Calculate ($X_{t+1} = sR + N_t + Y_{t+1} - W(Y_{t+1}) - \tilde{\Phi}\left[\tau(N_t + Y_t) - w(Y_t) - \widehat{W}_t\right]$) using the optimal $\widehat{W}_t$
   (d) Use the current iteration of the consumption function to solve for $C_{t+1}(X_{t+1})$
   (e) Use the EE to back out current period $C_t = u'^{-1}(\beta R \int_\nu u'(C_{t+1}) d\nu)$
   (f) Use CoH LOM to calculate $X_t = C_t + S_t + \widehat{W}_t$
2. Step 2: Solve for $\hat{W}$ when $S = 0$
   (a) Specify a grid for X_t from 0 up until the minimum X_t solved in Step 1
   (b) Use the FOC for $\widehat{W}_t$ to solve for the optimal $\widehat{W}_t$ assuming $S = 0$
   (c) Conditional on X_t and $\widehat{W}_t$, back out what C_t will be
3. Step 3: Iterate until the consumption function $C(X_t)$ converges

G Appendix – Estimating the Parameters of the Income Process

The following equations derive expressions for each of our income parameters as functions of the theoretical moments. Upper case variables represent annual variables and lower case variables represent bi-weekly variables. The theoretical moments are then estimated using sample moments calculated from within-individual variation across time. Lastly, the model parameters are calculated from the individual-level parameters by averaging across individuals.

$\alpha_{i,P}$

$$\mathbb{E}[p_{i,t,b}] = \mathbb{E}\left[\frac{P_{i,t}}{26}\right] + \mathbb{E}[\epsilon^P_{i,t,b}] \tag{3}$$

$$\alpha_{i,P} = \mathbb{E}[p_{i,t,b}]26 \tag{4}$$

$$\bar{\alpha}_{i,P} = \frac{\sum_t p_{i,t,b}}{T}26 \tag{5}$$

$$\bar{\alpha}_P = \frac{\sum_i \bar{\alpha}_{i,P}}{N} \tag{6}$$

σ^2_{i,ϵ^P}

$$p_{i,t,b} - \bar{p}_{i,t} = \epsilon^P_{i,t,b} \tag{7}$$

$$\sigma^2_{i,\epsilon^P} = \mathbb{V}[\epsilon^P_{i,t,b}] \tag{8}$$

$$\overline{\sigma^2}_{i,\epsilon^P} = \frac{\sum_t (\epsilon^P_{i,t,b})^2}{T} \tag{9}$$

$$\overline{\sigma^2}_{\epsilon^P} = \frac{\sum_i \overline{\sigma^2}_{i,\epsilon^P}}{N} \tag{10}$$

σ^2_{i,ν^P}

$$\mathbb{V}[p_{i,t,b}] = \mathbb{V}\left[\frac{P_{i,t}}{26}\right] + \mathbb{V}[\epsilon^Y_{i,t,b}] \tag{11}$$

$$\mathbb{V}[p_{i,t,b}] = \frac{\sigma^2_{i,\nu^P}}{26^2} + \sigma^2_{i,\epsilon^P} \tag{12}$$

$$\overline{\sigma^2}_{i,\nu^P} = \left(\frac{\sum_t (p_{i,t,b})^2}{T} - \overline{\sigma^2}_{i,\epsilon^P}\right)26^2 \tag{13}$$

$$\overline{\sigma^2}_{\nu^P} = \frac{\sum \sigma^2_{i,\nu^P}}{N} \tag{14}$$

$\alpha_{i,NP}$

$$\mathbb{E}[np_{i,t,b}] = \mathbb{E}\left[\frac{NP_{i,t}}{26}\right] \tag{15}$$

$$\alpha_{i,NP} = \mathbb{E}[np_{i,t,b}]26 \tag{16}$$

$$\bar{\alpha}_{i,NP} = \frac{\sum_t np_{i,t,b}}{T}26 \tag{17}$$

$$\bar{\alpha}_{NP} = \frac{\sum_i \bar{\alpha}_{i,NP}}{NP} \tag{18}$$

$\sigma^2_{i,\epsilon^{NP}}$

$$np_{i,t,b} - \overline{np}_{i,t} = \epsilon^{NP}_{i,t,b} \tag{19}$$

$$\sigma^2_{i,\epsilon^{NP}} = \mathbb{V}[\epsilon^{NP}_{i,t,b}] \tag{20}$$

$$\overline{\sigma^2}_{i,\epsilon^{NP}} = \frac{\sum_t (\epsilon^{NP}_{i,t,b})^2}{T} \tag{21}$$

$$\overline{\sigma^2}_{\epsilon^{NP}} = \frac{\sum_i \overline{\sigma^2}_{i,\epsilon^{NP}}}{N} \tag{22}$$

$\sigma^2_{i,\nu^{NP}}$

$$\mathbb{V}[np_{i,t,b}] = \mathbb{V}\left[\frac{NP_{i,t}}{26}\right] + \mathbb{V}[\epsilon^{NP}_{i,t,b}] \tag{23}$$

$$\mathbb{V}[np_{i,t,b}] = \frac{\sigma^2_{i,\nu^{NP}}}{26^2} + \sigma^2_{i,\epsilon^{NP}} \tag{24}$$

$$\overline{\sigma^2}_{i,\nu^{NP}} = \left(\frac{\sum_t (np_{i,t,b})^2}{T} - \overline{\sigma^2}_{i,\epsilon^{NP}}\right) 26^2 \tag{25}$$

$$\overline{\sigma^2}_{\nu^{NP}} = \frac{\sum \sigma^2_{i,\nu^{NP}}}{N} \tag{26}$$

H Appendix – Machine Learning Algorithm

Most transactions in the data do not contain direct information on spending category types. However, category types can be inferred from existing transaction data. In general, the mapping is not easy to construct. If a transaction is made at "McDonalds," it's easy to surmise that the category is "Fast Food Restaurants." However, it is much harder to identify smaller establishments such as "Bob's store." "Bob's store" may not uniquely identify an establishment in the data and it would take many hours of work to look up exactly what types of goods these smaller establishments sell. Luckily, the merchant category code (MCC) is observed for two account providers in the data. MCCs are four digit codes used by credit card companies to classify spending and are also recognized by the U.S. Internal Revenue Service for tax reporting purposes. If an individual uses an account provider that provides MCC information "Bob's store" will map into a spending category type.

The mapping from transaction data to MCC can be represented as $Y = f(X)$ where

Y represents a vector of MCC codes and X represents a vector of transactions data. The data is partitioned into two sets based on whether Y is known or not.[7] The sets are also commonly referred to as training and prediction sets. The strategy is to then estimate the mapping $\hat{f}(\cdot)$ from (Y_1, X_1) and predict $\hat{Y}_0 = \hat{f}(X_0)$.

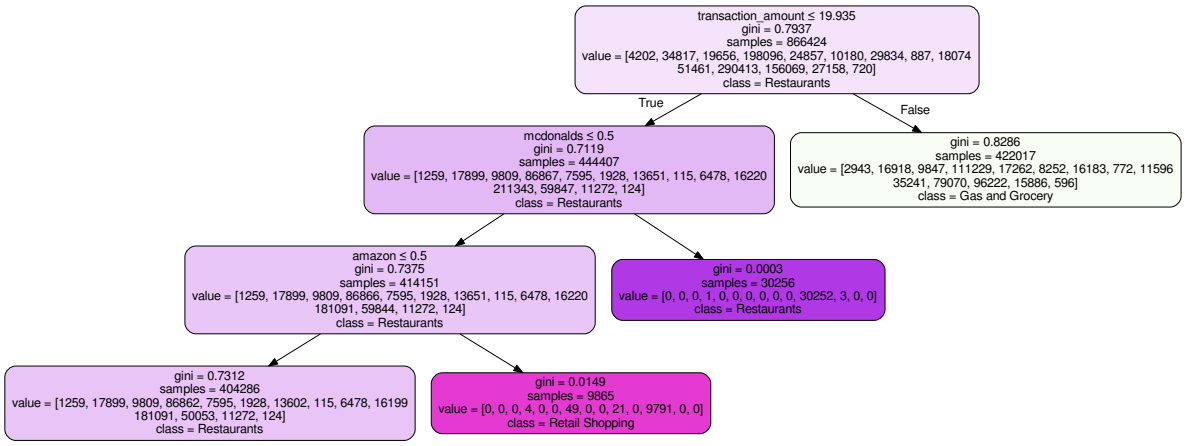
One option for the mapping is to use the multinomial logit model since the dependent variable is a categorical variable with no cardinal meaning. However, this approach is not well suited to textual data because each word would need its own dummy variable. Furthermore, interactions may be important for classifying spending categories. For example "jack in the box" refers to a fast food chain while "jack s surf shop" refers to a retail store. Including a dummy for each word can lead to about 300,000 variables. Including interaction terms will cause the number of variables to grow exponentially and will typically be unfeasible to estimate.

In order to handle the textual nature of the data we use a machine learning algorithm called random forest. A random forest model is composed of many decision trees that map transaction data to MCCs. This mapping is created by splitting the sample up into nodes depending on the features of the data. For example, for transactions that have the keyword "McDonalds" and transaction amounts less that $20, the majority of the transactions are associated with a MCC that represents fast food. To better understand how the decision tree works, Figure H.1 shows an example. The top node represents the state of the data before any splits have been made. The first row "transaction_amount $\leq$ 19.935" represents the splitting criteria of the first node. The second row is the Gini measure which is explained below. The third row shows that there are 866,424 total transactions to be classified in the sample. The fourth row "value=[4202,34817,...,27158,720]" shows the number of transactions in each spending category. The last row represents the majority class in this node. Because "Restaurants" has the highest number of transactions, assigning a random transaction to this category minimizes the categorization error without knowing any information about the transaction. At each node in the tree, the sample is split based on a feature. For example, the first split will be based on whether the transaction amount is $\leq$ 19.935. The left node represents all the transactions for which the statement is true and vice versa. Transactions

[7] Y_0 represents the set where Y is not known and Y_1 represents the set where Y is known.

≤ 19.935 are more likely to be "Restaurant" spending while transactions > 19.934 are more likely to be "Gas and Grocery." In our example, the sample is split further to the left of the tree. Transactions with the string "mcdonalds" are virtually guaranteed to be "Restaurant" spending. A further split shows that the string "amazon" is almost perfectly correlated with the category "Retail Shopping." How does the algorithm decide which features to split the sample on? The basic intuition is that the algorithm should split the sample based on features that lead to the largest disparities in the different groups. For example, transactions that have the word "mcdonalds" will tend to split the sample into fast food and non-fast food transactions so it is a good feature to split on. Conversely, "bob" is not a very good feature to split on because it can represent a multitude of different types of spending depending on what the other features are.

Figure H.1: Decision tree example



We state the procedure more formally by adapting the notation used in (Pedregosa et al., 2011). Define the possible features as vectors $X_i \in R^n$ and the spending categories as vector $y \in R^l$. Let the data at node m be presented by Q. For each candidate split $\theta = (j, t_m)$ consisting of a feature j and threshold t_m, partition the data into $Q_{left}(\theta)$ and $Q_{right}(\theta)$ subsets so that

$$Q_{left}(\theta) = (X, y)|x_j \leq t_m \tag{27}$$

$$Q_{right}(\theta) = (X, y)|x_j > t_m \tag{28}$$

The goal is then to split the data at each node in the starkest way possible. A popular quantitative measure of this idea is called the Gini criteria and is represented by

$$H(X_m) = \sum_k p_{mk}(1 - p_{mk}) \tag{29}$$

where $p_{mk} = 1/N_m \sum_{x_i \in R_m} \mathbb{I}(y_i = k)$ represents the proportion of category k observations in node m.

If there are only two categories, the function is is minimized at 0 when the transactions are perfectly split into the two categories[8] and maximized when the transactions are evenly split between the two categories.[9]

Therefore, the algorithm should choose the feature to split on that minimizes the Gini measure at node m

$$\theta^* = argmin_\theta \frac{n_{left}}{N_m} H(Q_{left}(\theta)) + \frac{n_{right}}{N_m} H(Q_{right}(\theta)) \tag{30}$$

The algorithm acts recursively so the same procedure is performed on $Q_{left}(\theta^*)$ and $Q_{right}(\theta^*)$ until a user-provided stopping criteria is reached. The final outcome is a decision rule $\hat{f}(\cdot)$ that maps features in the transaction data to spending categories.

This example shows that decision trees are much more effective in mapping high dimensional data that includes text to spending categories. However, fitting just one tree might lead to over-fitting. Therefore, a random forest fits many trees by bootstrapping the samples of the original data and also randomly selecting the features used in the decision tree. With the proliferation of processing power, each tree can be fit in parallel and the final decision rule is based on all the decision trees. The most common rule is take the majority decision of all the trees that are fit.

Table H.1 shows our goodness of fit measures when we train the model on 70% of the

[8] because 0*1 + 1*0 = 0.

[9] because 0.5*0.5 + 0.5*0.5 = 0.5.

data and use the remaining 30% as the testing data set. We calculate the measures at the category level as well as our aggregated measure. The aggregate measure has higher precision and recall because a transaction is still coded as correct as long as it is identified as one of the four non-durable consumption categories. Accuracy can only be calculated for the aggregate measure.

Table H.1: Goodness of fit measures

	Precision	Recall	Accuracy	Share
Restaurants	0.92	0.94	-	0.51
Gas and Grocery	0.92	0.94	-	0.38
Entertainment	0.90	0.78	-	0.06
Misc. Services	0.92	0.77	-	0.05
Aggregate	0.96	0.96	0.95	1.00

Notes: Precision measures the fraction of predicted consumption transactions that are correctly predicted. Recall measures the fraction of actual consumption transactions that are correctly predicted. Accuracy calculates the fraction of total observations that are correctly predicted. The last column shows the share of transactions in each category.

I Appendix – Alternate parameter values

Table I.1: Average tax refund under different parameter values ($\theta = 4$)

		β				
		0.975	0.980	0.985	0.990	0.995
	0.30	1,267	1,292	1,318	1,343	1,366
	0.40	1,935	1,973	2,043	2,128	2,203
	0.50	2,722	2,761	2,811	2,884	2,947
γ	0.57	3,196	3,233	3,293	3,373	3,451
	0.60	3,490	3,531	3,591	3,676	3,764
	0.70	4,223	4,268	4,330	4,420	4,530
	0.80	5,118	5,154	5,194	5,312	5,437

Notes: This table calculates the average tax refund for 100,000 simulated observations under different parameter values.

References

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos,

D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, 2011, *12*, 2825–2830.