

# Intro

LendingClub is a platform for peer-to-peer lending. If you're not familiar with it, understand the following:

- **Peer-to-peer (P2P) lending:** It's crowdfunding peoples' loans and they are expected to pay back with interest.
- **Note:** A piece of a loan. If somebody wanted to borrow \$1000, a note could be a \$25 chunk of that.
- **Secondary market:** Where investors can buy or sell notes. Investors might sell for profit, or they might sell at a discount because they think there's too much risk in holding a note.
- **Charged off:** When a borrower stops paying his/her loan and there is no chance of recovering any money. This is bad for the investors.

Here's what I did:





- Downloaded existing loan data which LendingClub provides.
- Found patterns in that data to recognize a good or bad loan.
- Created script find notes based on my patterns.
- Created interface to buy those notes easily.

## Step 1. Get the dataset

### Extract

Every quarter, LendingClub updates a list of all loans they've issued and what the status of those loans are. It's a large dataset, with plenty of *interesting* fields. There are multiple csv files with rows about each loan such as loan amount, interest rate, purpose, date issued, current status, et al.

*The .csv files have 100k+ rows per file:*

Name	Date modified	Type	Size
 LoanStats3a_securev1.csv	3/1/2016 10:48 PM	OpenOffice.org 1....	33,965 KB
 LoanStats3b_securev1.csv	1/25/2016 4:18 PM	OpenOffice.org 1....	125,905 KB
 LoanStats3c_securev1.csv	1/25/2016 4:19 PM	OpenOffice.org 1....	142,312 KB
 LoanStats3d_securev1.csv	1/25/2016 4:20 PM	OpenOffice.org 1....	250,457 KB

*More than enough data for LibreOffice to handle on my computer (it's actually not responding in the screenshot below)*

member_id	loan_amount	funded_amount	term	int_rate	installment	grade	sub_grade	emp_title	emp_length	home_ownership	annual_inc	verification_status	issue_d	loan_status	purpose	int
40002027	15000	15000	15000 60 months	12.30%	326.64	C1	MANAGEMENT	10+ years	RENT	78000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=30006214	debt consolidation	Debt consolidation
30562024	10400	10400	10400 36 months	6.99%	321.05	A1	Truck Driver Delivery Personnel	8+ years	ACMPTGAGE	58000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=30005048	credit_card	Credit card refinanc
40562021	9600	9600	9600 36 months	13.00%	326.53	C3	Admin Specialist	10+ years	RENT	69000 Source Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862287	debt consolidation	Debt consolidation
40174472	12800	12800	12800 48 months	17.44%	339.90	D4	Senior Sales Professional	10+ years	RENT	129000 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862284	car	Car financing
40002024	21425	21425	21425 60 months	15.00%	334.30	D1	Programming Analyst Supervisor	6+ years	RENT	63000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842129	credit_card	Credit card refinanc
4043502	7600	7600	7600 36 months	13.60%	262.21	C2	Technical Specialist	< 1 year	RENT	90000 Source Verified	Dec-2014	Charged Off	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862254	debt consolidation	Debt consolidation
40474561	10000	10000	10000 36 months	12.99%	332.18	B5	Investment Consultant	8+ years	RENT	90000 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37702366	debt consolidation	Debt consolidation
40045008	5200	5200	5200 36 months	11.44%	172.88	B4	Store Manager	2+ years	RENT	26000 Not Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842122	debt consolidation	Debt consolidation
40045005	2500	2500	2500 36 months	11.99%	83.03	B1	Manufacturing Engineer	< 1 year	ACMPTGAGE	66000 Source Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37743281	home_improvement	Home improvement
40117199	16000	16000	16000 60 months	11.44%	351.48	B4	Foreign Service Officer	10+ years	OWN	100777 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842144	debt consolidation	Debt consolidation
40462027	21075	21075	21075 60 months	21.40%	361.80	E1	Paralegal	10+ years	ACMPTGAGE	60000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37712168	debt consolidation	Debt consolidation
30567179	23225	23225	23225 36 months	14.31%	803.71	C4	Teacher	10+ years	RENT	70000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=30464663	credit_card	Credit card refinanc
40562021	12975	12975	12975 36 months	17.44%	468.12	D6	Sales	10+ years	RENT	60000 Source Verified	Dec-2014	Loan (30-120 days)	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862122	home	Home buying
40453027	17000	17000	17000 36 months	13.00%	378.22	C3	Deputy Sheriff	10+ years	ACMPTGAGE	70000 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862228	debt consolidation	Debt consolidation
40501584	6000	6000	6000 36 months	10.49%	194.90	B3	Assistant manager	10+ years	ACMPTGAGE	120000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842126	home_improvement	Home improvement
36252618	3000	3000	3000 36 months	10.49%	87.18	B3	Respiratory Therapist	9+ years	RENT	60000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=36520173	medical	Medical expenses
40121717	2000	2000	2000 36 months	15.99%	70.12	C2	Student Services Coordinator	7+ years	RENT	37000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842142	other	Other
40501586	2000	2000	2000 36 months	14.99%	69.31	C9	janitor plan associate	8+ years	RENT	32000 Not Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37742142	credit_card	Credit card refinanc
38012192	18000	18000	18000 36 months	10.99%	537.70	B5	Education Admining Manager	8+ years	OWN	70000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=36306613	debt consolidation	Debt consolidation
36121308	7200	7200	7200 36 months	12.29%	241.45	C1	Director of Operations	5+ years	ACMPTGAGE	84000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=36442163	debt consolidation	Debt consolidation
40511312	13500	13500	13500 60 months	18.99%	355.80	E3	nurse	5+ years	OWN	40000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37752129	debt consolidation	Debt consolidation
40373014	14000	14000	14000 36 months	10.49%	444.47	B3	High grade teacher	10+ years	OWN	42000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37761266	credit_card	Credit card refinanc
40502015	6000	6000	6000 36 months	14.99%	207.87	C5	office manager	1 year	RENT	20000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37742139	debt consolidation	Debt consolidation
40502029	10800	10800	10800 36 months	6.49%	345.18	B2	Asset Protection	8+ years	ACMPTGAGE	42000 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37761266	debt consolidation	Debt consolidation
40551175	4000	4000	4000 36 months	15.99%	145.62	D2	Accounting clerk	9+ years	RENT	50000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37852069	debt consolidation	Debt consolidation
38012198	2000	2000	2000 36 months	12.99%	67.84	C3		n/a	RENT	21112 Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=36310199	medical	Medical expenses
40380055	10000	10000	10000 60 months	20.99%	405.72	E4	Appraiser	3+ years	ACMPTGAGE	48000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37822029	other	Other
40501710	18400	18400	18400 36 months	14.14%	633.86	C4	Construction Foreman	10+ years	ACMPTGAGE	108000 Not Verified	Dec-2014	Loan (30-120 days)	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37822026	home_improvement	Home improvement
40545000	2000	2000	2000 36 months	14.14%	66.60	C4	Sales Rep	2+ years	RENT	52000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37822028	other	Other
40502060	1000	1000	1000 36 months	12.29%	13.41	C1	Attorney	3+ years	OWN	60000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37822025	other	Other
40420018	8000	8000	8000 36 months	10.49%	162.49	B1	Teacher	4+ years	ACMPTGAGE	52000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862149	debt consolidation	Debt consolidation
40401191	26000	26000	26000 60 months	10.49%	601.78	B3	Hortana Cams Manager	3+ years	ACMPTGAGE	61000 Source Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862106	debt consolidation	Debt consolidation
40404055	20075	20075	20075 60 months	12.29%	470.70	C1	Prophet Jailer	10+ years	RENT	48000 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842166	home_improvement	Home improvement
40544084	12000	12000	12000 60 months	17.80%	303.82	D6	Nurse	4+ years	RENT	60000 Source Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842166	debt consolidation	Debt consolidation
40541586	4000	4000	4000 36 months	10.49%	149.49	B3	Production Assistant	8+ years	RENT	40000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862111	debt consolidation	Debt consolidation
40544596	8000	8000	8000 36 months	12.29%	287.21	C1	Diner	8+ years	RENT	70000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37861688	debt consolidation	Debt consolidation
40414005	8025	8025	8025 36 months	20.99%	239.50	D1	Graduate Teaching Assistant	2+ years	RENT	28000 Not Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862123	debt consolidation	Debt consolidation
40545098	5000	5000	5000 36 months	12.99%	168.45	C2	engineer	10+ years	RENT	96000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862013	house	Home buying
40471008	2400	2400	2400 36 months	14.14%	82.30	C4	log	7+ years	OWN	30000 Source Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37762013	other	Other
40411809	13000	13000	13000 60 months	11.99%	333.18	B3	Financial Advisor	7+ years	RENT	40000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37762011	debt consolidation	Debt consolidation
40502010	5000	5000	5000 36 months	13.00%	171.07	C3	Accounting Manager	10+ years	RENT	54707 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37762071	moving	Moving and relocation
40504040	8025	8025	8025 60 months	16.49%	138.81	D1	Personal Advisor	< 1 year	ACMPTGAGE	76000 Not Verified	Dec-2014	Fully Paid	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842162	debt consolidation	Debt consolidation
40544174	8000	8000	8000 36 months	7.49%	298.38	A4	Store Manager	7+ years	ACMPTGAGE	43000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37861488	debt consolidation	Debt consolidation
40404056	1000	1000	1000 36 months	11.99%	151.18	B1	teacher	10+ years	ACMPTGAGE	49000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862111	other	Other
40544032	10000	10000	10000 36 months	10.49%	324.48	B3	R1 Director	10+ years	RENT	127000 Source Verified	Dec-2014	Charged Off	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37861629	debt consolidation	Debt consolidation
40404052	14000	14000	14000 60 months	14.49%	400.00	D1	Customer Support	9+ years	RENT	40000 Source Verified	Dec-2014	Charged Off	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37762161	credit_card	Credit card refinanc
40541122	33000	33000	33000 60 months	12.29%	778.11	C1	Manager, Information Technology	10+ years	OWN	133995 Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37862066	credit_card	Credit card refinanc
40414084	3100	3100	3100 36 months	14.14%	82.11	C4	Sign Artist	9+ years	RENT	53000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37842161	debt consolidation	Debt consolidation
40375021	6000	6000	6000 36 months	11.99%	288.26	B5	International Advisor	2+ years	RENT	42000 Not Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37812112	debt consolidation	Debt consolidation
40441428	21000	21000	21000 60 months	8.10%	427.72	A5	Male Lab Manager	9+ years	ACMPTGAGE	40000 Source Verified	Dec-2014	Current	n	https://www.lendingclub.com/showLoanDetail.action?loan_id=37720951	debt consolidation	Debt consolidation

### Load

Having worked with the LAMP (Linux, Apache, MySQL, PHP) stack, I recognized that a few hundred thousand rows and many numerical columns would be a good fit for a relational database like MySQL. However at time, MongoDB was quite popular and I wanted to see what the hype was about. This was my opportunity to try it! **[Lesson to be learned here.]**

After getting MongoDB set up on my computer, I was ready to import the data in the csv files. There is a tool called mongoimport to do this. **Because LendingClub updates the csv files once a quarter, I expected the need to have to redo this step, so I wrote down the steps I took.** When I needed to import data again a few months later, I was so happy I did.

```
2  import steps:
3      Download all CSV's
4      Delete first line in each CSV
5      Clear out the loans database
6      Run command for each csv:
7      D:\Mongo\bin\mongoimport.exe /h localhost /p 27017 /d lc /c loans /numInsertionWorkers 2 /type csv /headerline /fi
```

## Clean (Transform)

Below is an example of a row from the csv files. Some fields didn't look necessary, and some needed to be converted to a proper type. Using an appropriate data type can save space and can make tasks such as sorting and searching faster.

```
"1069410","1303652","21000","21000","20975"," 60 months","
19.91%","555.33","E","E4","Costco","7 years","RENT","50000","Verified","Dec-2011","Charged
Off","n","https://www.lendingclub.com/browse/loanDetail.action?loan_id=1069410","
Borrower added on 12/20/11 > consolidation of credit cards.<br>","debt_consolidation","
Bill pay
of","980xx","WA","21.58","0","Sep-1998","680","684","1","","","7","0","19448","97.6%","14"
,"f","0.00","0.00","18319.14","18297.35","8990.81","9328.33","0.0","0.0","0.0","Oct-2014",
"555.33","","Jan-2016","634","630","0","","1","INDIVIDUAL","","","0","","","","","",
","","","","","","","","",""
```

- URL isn't necessary. With just a loan id, it can be reconstructed.
- Some dates should be converted from string.
- Interest rates should be converted from string.

This can be done directly in MongoDB with Javascript. I put a number of code blocks like below into "post\_processing.js" so I can run them all each time I import data. Although convenient, it's slow. It would have been a better to clean first, then load.

```
1.  /*
2.  Fix dates
3.  */
4.  db.loans.find({
5.      issue_d: {
6.          $type: 2
7.      }
8.  }).forEach( function( elem ) {
9.      var newDate = elem.issue_d.replace( '-', ' 1 ' );
10.     elem.issue_d = new Date( newDate );
11.     db.loans.save( elem );
12. });
```

## Step 2. “Analysis”

My goal here was to find patterns that classify a loan as good or bad. A good loan being one that is fully paid off, and a bad loan being one that is late, in default, or charged off. My strategy was to pick values of a data columns(*features*) and see what the ratio good loans:bad loans was, and compare that to the average ratio of good:bad.

In pseudocode, this is what I did:

```
For loans with purpose “small_business”:
    if the loan is bad:
        numerator++
    total++
Ratio_feature = numerator / total

Reset numerator and total

For all loans:
    If the loan is bad:
        numerator++
    Total++
Ratio_average = numerator / total
```

I then expanded the number of features and their values and calculated ratios for each combination. It’s far from elegant.

```
1. var home_ownership_steps =
2. {
3.     "0" : "RENT",
4.     "1" : "MORTGAGE",
5.     "2" : "OWN",
6.     "3" : "NONE",
7.     //"4" : "OTHER",
8.     //"5" : "ANY"
9. };
10.
11.
12. for( var emp_length_key in emp_length_steps )
13. {
14.     for( var verification_status_key in verification_status_steps )
15.     {
16.         for( var purpose_key in purpose_steps )
17.         {
18.             for( var home_ownership_key in home_ownership_steps )
19.             {
20.                 for( var income_min = annual_inc_start; income_min <= annual_inc_end; income_min +=
annual_inc_step_size )
21.                 {
```

I even tried language processing (if you’d be kind enough to allow me to call it that). In the loan description, did borrowers use lowercase/capital “i” when referring to themselves? Were there any keywords that could be red flags?

```
1. var descriptionsToTry = [
2.     /.*/i ,
3.     // /\.s+[A-Z]/, // Capital letter after a period...actually worse in a lot of cases
4.     // /improve/i,
5.     // /credit/i,
6.     // /smart/i,
7.     // /late/i,
8.     // /history/i
9.     // /soon/i
```

```

10. //
11. ];
12.
13. for( var key in descriptionsToTry ) {
14.     var description = descriptionsToTry[key];
15.
16.     var numerator = db.loans.count({
17.         desc: description
18.     })

```

I didn't reach any solid conclusions from this, but it was worth trying. With more sophisticated language processing techniques, I think I would have had a better chance of finding actionable patterns.

## MapReduce

MapReduce is a way to filter and process a large amount of records in parallel. With each record in your dataset, you apply some function (**map**) to transform/preprocess that record. Then, you output(emit) a key and value for each record. Finally, everything with the same key gets grouped together and processed in the **reduce** stage.

I was able to use MapReduce to do my "analysis" in a much cleaner, faster way. In my map function, I emitted a key of all the features I thought would be important in guessing if a borrower was "good" or "bad." The value emitted contained whether or not that loan was bad, and a constant "1" used to count the number of loans with the same key. The reduce function simply added up the counts.

```

1. // Map function
2. var key = {
3.     emp_length: this.emp_length,
4.     verification_status: this.verification_status,
5.     purpose: this.purpose,
6.     annual_inc: roundDown(this.annual_inc, 20000),
7.     home_ownership: this.home_ownership,
8.     fico_range_low: roundDown(this.fico_range_low, 50),
9.     total_acc: roundDown(this.total_acc, 10),
10.    revol_util: roundDown(this.revol_util, 10),
11.    loan_amnt: roundDown(this.loan_amnt, 3000)
12. };
13.
14. var bad = 0;
15. if (this.loan_status == "Default" || this.loan_status == "Charged Off") {
16.     bad = 1;
17. }
18.
19. emit(key, {
20.     count_bad: bad,
21.     count_total: 1
22. });
23.
24.
25. // Reduce function
26. reducedVal = {
27.     count_bad: 0,
28.     count_total: 0
29. };
30.
31. for (var idx = 0; idx < array_values.length; idx++) {
32.     reducedVal.count_bad += array_values[idx].count_bad;
33.     reducedVal.count_total += array_values[idx].count_total;
34. }
35.
36. return reducedVal;

```

Below is what the result of my MapReduce looks like. `_id` is the key I emitted.

```
1.  /* 1 */
2.  {
3.    "_id" : {
4.      "emp_length" : "1 year",
5.      "verification_status" : "Not Verified",
6.      "purpose" : "car",
7.      "annual_inc" : 0,
8.      "home_ownership" : "RENT",
9.      "fico_range_low" : 650,
10.     "total_acc" : 0,
11.     "revol_util" : 30,
12.     "loan_amnt" : 3000
13.   },
14.   "value" : {
15.     "bad" : 0,
16.     "total" : 1,
17.     "ratio" : 0
18.   }
19. }
20.
21. /* 2 */
22. {
23.   "_id" : {
24.     "emp_length" : "1 year",
25.     "verification_status" : "Not Verified",
26.     "purpose" : "car",
27.     "annual_inc" : 0,
28.     "home_ownership" : "RENT",
29.     "fico_range_low" : 650,
30.     "total_acc" : 0,
31.     "revol_util" : 50,
32.     "loan_amnt" : 3000
33.   },
34.   "value" : {
35.     "bad" : 0,
36.     "total" : 1,
37.     "ratio" : 0
38.   }
39. }
```

Later, I can use this collection which I called “person\_analysis” to estimate risk for a loan. If I’m happy with the ratio, I’ll buy a note for that loan.

## Step 3. Scrape secondary market loans

The secondary market allows investors to buy or sell existing notes

- (1) Typically there are hundreds of thousands of notes available on the secondary market.
- (2) FolioFN Provides a way to filter all these notes.
- (3) You can then download a csv file containing only filtered notes or all notes on the secondary market (400k in the screenshot)

Below are the notes listed for sale. The notes belong to either existing members and are held in their existing escrow accounts.

1

403,064 of 492,788 notes

Add to Order

Filters

Less

X

Interest Rate

From: 4.0% To: 29.0%

Remaining Payments

From: 1 To: 60

Asking Price

From: 0.00 To: Any

Loan Term

☒ 36 months
 ☒ 60 months

Recent Credit Score

600 or Less

850

≤600 - 850

Yield to Maturity

From: Any % To: Any %

Outstanding Principal

From: 0.00 To: Any

Credit Score Change

☒
☒
☒

Markup / Discount

100% Discount

70+% Markup

(100%) - 15%

Loan Status

☒ Issued
 ☐ Late 16-30
 ☒ Current
 ☐ Late 31-120
 ☐ In Grace
 ☒ Never Late

Original Note Amount

From: 25 To: Any

Exclude Loans

☐ Exclude Loans I have already invested in

Reset

Apply

2

NOTE ID	AMOUNT	INTEREST RATE	TERM	STATUS	MARKUP / DISCOUNT	REMAINING PAYMENTS	CREDIT SCORE
660-664	\$25	10.99%	36	Current		4	5

Add to Order

3

[Download Search Results](#)
[Download Full Inventory](#)

Using “Download Full Inventory” is easy, but it can show notes from loans I already invested in. I don’t want to buy multiple notes from the same loan because it increases the penalty if that loan defaults. At the time I worked on this, the Foliofn API didn’t exist so I had to go to this page each day, apply my filters, and then download the results. After maybe 2 days I was sick of it so I decided to automate.

PhantomJS lets me pretend to be a user clicking things in a browser (headless browser). CasperJS makes using PhantomJS easier, and is what I used to accomplish this task.

```

1. casper.then( function() {
2.   this.capture( 'filled filters.png' );
3. });
4.
5. casper.thenClick( "#filter-body input[value=Apply]" );
6.
7. casper.wait( 2000, function() {
8.   this.capture( 'filtered results.png' );
9.   console.log( "Downloading notes" );
10.  casper.download( casper.getElementInfo( '#notesDownloadSearchResultsLink' )['attributes']['href'],
    'browseNotes.csv' );
11. })

```

## Step 4. Find and buy good notes

### Loan buying backend

At this point I have

1. A way to get notes that are currently for sale
2. A database collection of existing loans
3. A database collection of predicted risk

So I put them all together into a PHP script.

First run the CasperJS script to download the notes for sale for me.

```
1. system( 'casperjs --ssl-protocol=tlsv1 C:\casperjs\bin\lendingclub.js' );
```

Then, I load all those notes into a temporary MongoDB collection called *notes*. I could've just parsed the csv lines in PHP but I wanted to do it with MongoDB.

```
1. system( 'D:\Mongo\bin\mongoimport.exe /h localhost /p 27017 /d lc /c notes /numInsertionWorkers 2 /type
  csv /headerline C:\xampp\htdocs\lc\browseNotes.csv' );
```

Next I go through each of these notes for sale, and look up the original loan data using LoanId.

```
1. $loanInMongo = $loansDB->findOne(
2.     json_decode( '
3.         {
4.             "id": ' . $folioNote['LoanId'] . '
5.         }
6.     ')
7. );
```

If I get a match, I'll run the loan data against my predicted risk collection and add the risk to that note data. Then, I can go back and query all the notes that have an estimated risk and filter by some maximum risk and some minimum sample size. I print the result in JSON for the frontend to display.

```
1. $notesDB->find(
2.     Array(
3.         "ratio" => Array (
4.             '$lt' => .08
5.         ),
6.         "total" => Array (
7.             '$gt' => 5
8.         )
9.     )
10. )
11. );
```



# Loan buying front-end

I used DataTables in the frontend to display the notes for the ability to sort columns. I also added a column with links to Open (see more details about the note) or Add to order (add to cart).

Login page opened
Found email field, filling in values
In account page!
In Failo Account
In browse tradable notes
In browse notes
Filling in filters
Downloading notes
4653 notes imported

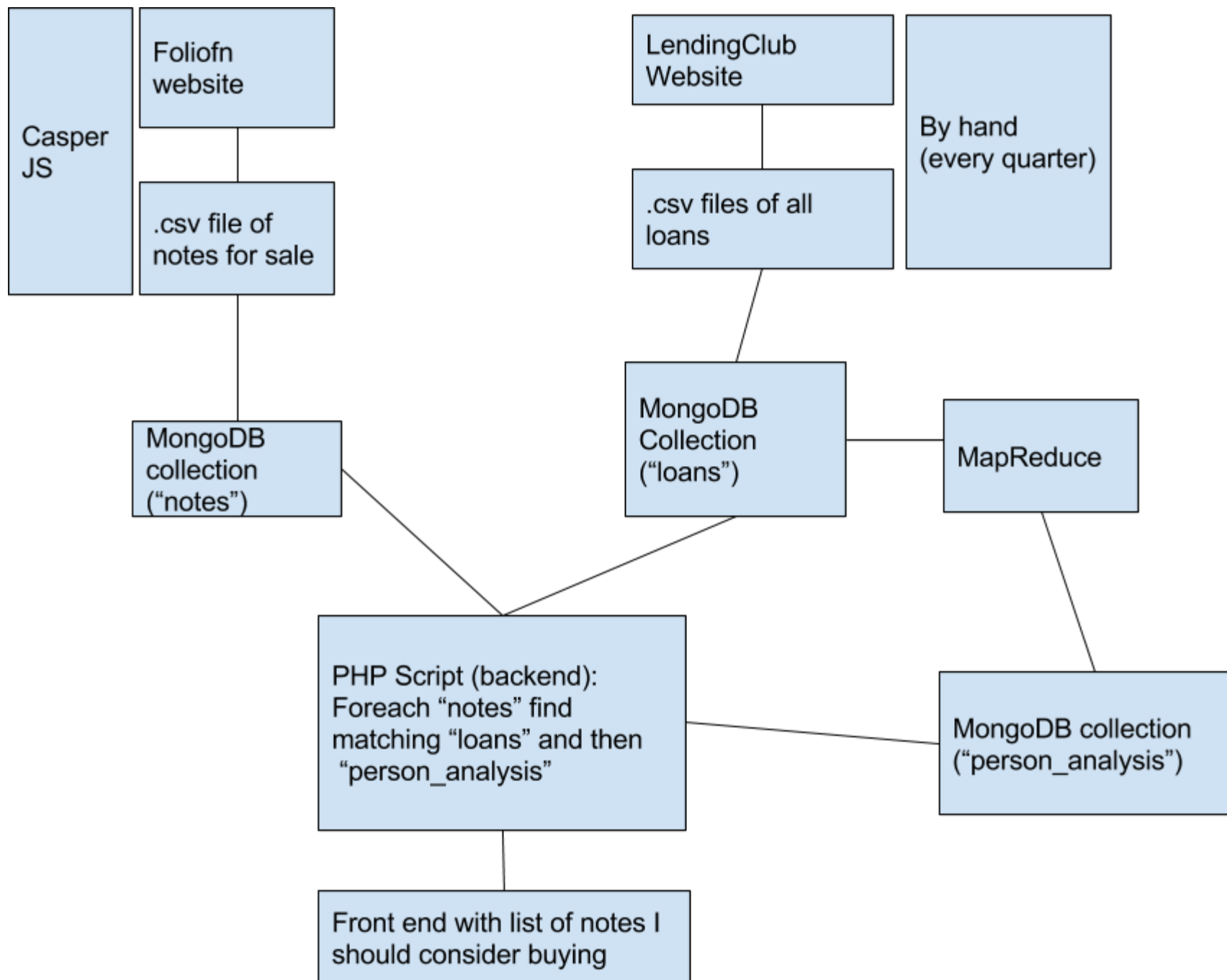
Show 25 entries

ratio	bad	total	annual_inc	emp_length	verification_status	purpose	home_ownership	fico_range_low	total_acc	revol_util	loan_amnt	Remaining Payments	DaysSinceLastPayment	AskPrice	YTM	Markup/Discount
0	0	6	38000	3 years	Not Verified	credit_card	RENT	695	10	40.1	8000	10	18	7.68	7.68	-0.66
0	0	14	39000	2 years	Not Verified	debt_consolidation	RENT	695	12	89.4	7000	13	10	10.27	10.54	0.23
0	0	8	40000	4 years	Not Verified	debt_consolidation	RENT	680	14	67.2	12000	9	15	7.12	5.88	0.52
0	0	7	44500	7 years	Not Verified	debt_consolidation	RENT	660	23	62.9	8000	9	22	7.53	11.56	0.22
0	0	8	39000	8 years	Not Verified	credit_card	RENT	695	18	56.1	6000	8	8	13.27	5.12	1.7
0	0	10	24000	5 years	Not Verified	debt_consolidation	RENT	690	16	53.5	10000	10	15	16.29	5.12	1.9
0	0	6	40000	2 years	Not Verified	debt_consolidation	RENT	665	10	65.7	15000	6	22	5	3.5	0.92
0	0	7	60509.73	6 years	Not Verified	debt_consolidation	RENT	675	27	66.3	6000	9	22	7.03	3.97	0.56
0	0	6	53000	2 years	Not Verified	debt_consolidation	RENT	700	18	52.4	12500	9	3	7.12	5.12	0.91
0.076923076923077	1	13	35000	2 years	Not Verified	credit_card	RENT	675	18	81.7	8875	13	4	10.8	4.03	4.58
0.076923076923077	1	13	55000	4 years	Not Verified	debt_consolidation	RENT	680	17	71.7	11000	11	22	26.04	8.91	0.01

Showing 1 to 11 of 11 entries



## Block Diagram



## Along came Pandas

The Mongo+PHP+CasperJS system was great for a while. But then I found Pandas, a library for Python that makes data munging and analysis much more convenient. I didn't write about all of it here, but it's on GitHub (<https://github.com/dharik/Lending-Club>).

## Lessons Learned

Use the best tool for the job.

Early on, I recognized that the loan data was a good fit for a relational database but I instead used MongoDB. It was nice to learn and use MongoDB as well as get experience with MapReduce.

Now, I could have used MongoDB in a way that it's better suited, such as putting each note as a subdocument for the loan it belongs to. But I stopped due to time and LendingClub having some big problems.

## Use version control.

While I did my “analysis,” all along the way I was modifying my files and not saving copies of them anywhere. There was some good stuff I found that I ended up erasing just because I wanted to try something else. It would've helped to be able to go back to a previous state.

## Recognize Terminology

Many of the steps I performed on my own intuition have names, which I tried to incorporate in my writing. For example, my “Step 1” was an Extract, Transform, Load (ETL) pipeline. Knowing what the process I'm working on is called can make it easier to study and learn.