Project Child Support

Code ▾

Saturday, March 4, 2017

The data for this project describe payments for child support made to a government agency. A "case" refers to a legal judgment that an absent parent (abbreviated in variable names as "AP") must make child support payments. The data is distributed in *four* CSV files, stored on in the data folder. The data are distributed "as is" as obtained from the agency (albeit anonymized). Most categorical variables are self-explanatory.

The file **cases.csv** has six columns, one for each case:

- CASE_NUM Unique case identifier
- CASE_STATUS ACV (active), IN_ (inactive), IC_ (closed), IO_ (legal), IS_(suspend)
- CASE_SUBTYPE AO (arrears), EF (foster), MA (medical), NO (arrears), RA (regular), RN (regular)
- CASE_TYPE AF (AFDC), NA (non-afdc), NI (other)
- AP_ID Identifying number for absent parent
- LAST_PYMNT_DT Recorded date of last payment

The file **parents.csv** has 10 columns, one for each parent:

- AP_ID Unique identifier for parent
- AP_ADDR_ZIP Coded na for missing, 0 for "known unknown", 1 for city, 2 south state, 3 north state, 4 other
- AP_DECEASED_IND AP is deceased
- AP_CUR_INCAR_IND AP is incarcerated
- AP_APPROX_AGE
- MARITAL_STS_CD Self-explanatory
- SEX_CD
- RACE_CD Categorical
- PRIM_LANG_CD Language code
- CITIZENSHIP_CD Citizenship code

The file **children.csv** has 9 columns:

- CASE_NUM Case number
- ID Unique identifier for child
- SEX_CD
- RACE_CD
- MARITAL_STS_CD Marital status code
- PRIM_LANG_CD Primary language
- CITIZENSHIP_CD
- DATE_OF_BIRTH_DT
- DRUG_OFFNDR_IND Past drug offence

The file **payments.csv** has only six columns, but more than 1.5 million records:

- CASE_NUM Case number for the payment
- PYMNT_AMT Dollar amount of payment
- COLLECTION_DT Date of payment
- PYMNT_SRC A (regular), C (worker comp), F (tax offset), I (interstate), S (st tax), W (garnish)
- PYMNT_TYPE A (cash), B (bank), C (check), D (credit card), E (elec trans), M (money order)
- AP_ID Absent parent ID

1. File linkage integrity

Installing necessary packages, loading the data, and viewing the dimensions:

Hide

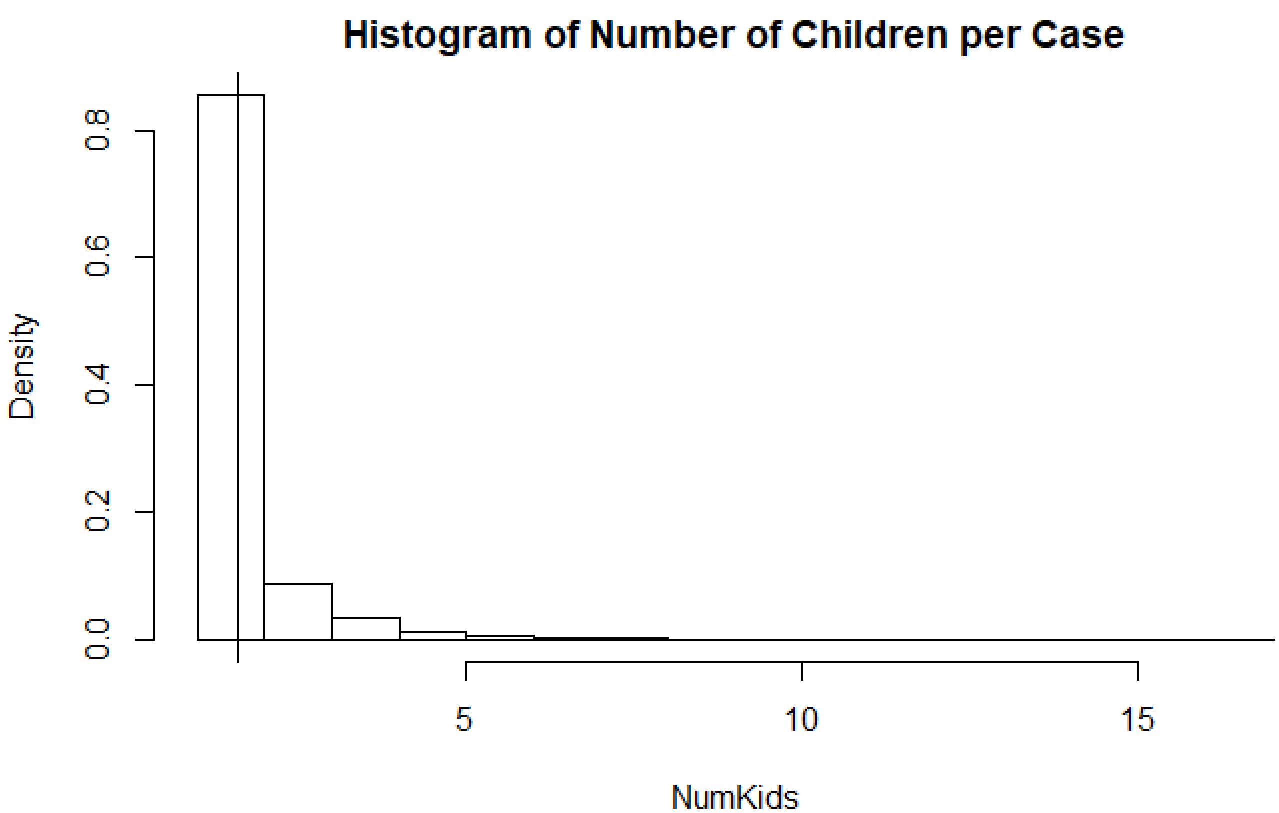
```
library("dplyr")
library("ggplot2")
# Load data into data frames. Change path to the location of your data files
path <- "C:/Users/Calle/Downloads/STAT405-Payments/"
cases <- read.csv(paste0(path, "cases.csv"))
```

```
cannot open file 'C:/Users/Calle/Downloads/STAT405-Payments/cases.csv': No such file or director
yError in file(file, "rt") : cannot open the connection
```

Examining histogram and average number of children per case:

Hide

```
#Histogram
NumKids <- as.vector(table(children$CASE_NUM))
hist(NumKids, prob=TRUE, main="Histogram of Number of Children per Case")
#Mark the location of the mean
abline(v=mean(NumKids))
```



There is an average 1.6 children per case.

Maximum number of cases per child:

Hide

```
table(children$ID)[which.max(as.vector(table(children$ID)))]
```

```
153343287
       12
```

Hide

```
library(dplyr)
```

```


Attaching package: <U+393C><U+3E31>dplyr<U+393C><U+3E32>

The following objects are masked from <U+393C><U+3E31>package:stats<U+393C><U+3E32>:

    filter, lag

The following objects are masked from <U+393C><U+3E31>package:base<U+393C><U+3E32>:

    intersect, setdiff, setequal, union
```

Hide

```
filter(children, ID=="153343287")
```

CASE_NUM <int>	ID <int>	SEX... <fctr>	RAC... <fctr>	MARITAL_STS... <fctr>	PRIM_LANG... <fctr>	CITIZENSHIP_CD <fctr>	DATE_(<fctr>
881385019	153343287	M	B	N		C	4/12/20
901516566	153343287	M	B	N		C	4/12/20
991517158	153343287	M	B	N		C	4/12/20
1041455891	153343287	M	B	N		C	4/12/20
221411290	153343287	M	B	N		C	4/12/20
341399284	153343287	M	B	N		C	4/12/20
371506852	153343287	M	B	N		C	4/12/20
411385706	153343287	M	B	N		C	4/12/20
611508498	153343287	M	B	N		C	4/12/20
631399690	153343287	M	B	N		C	4/12/20

1-10 of 12 rows | 1-8 of 9 columns Previous **1** 2 Next

The maximum number of cases attached to any given child is 12 cases

Making sure every absent parent (AP_ID) identified in the payments data have an identifying record in the parents data file

Hide

```
test <- subset(payments, !(payments$AP_ID %in% parents$AP_ID))
test
```

```
0 rows
```

Result: 0 rows, indicating that every payment has a corresponding parent in the payments dataframe

2. Recoding categories

Some categorical variables among these data frames are sparse (seldom observed). For example, the variable PYMNT_SRC in Payments has category 'M' with 2 cases and category 'R' with 7. These are too few for modeling in regression.

For that reason, I will write a function pool_categories(data, threshold) that pools sparse categories with few occurences into an "*Other*" category and counts the frequency of categories within a given variable

Hide

```
pool_categories <- function(data,threshold) {
    i <- table(data) < threshold
    below_threshold <- names(table(data))[i]
    if ( "_Other_" %in% names(table(data)) ) { stop("Factor level '_Other_' already exists") }
    src <- as.character(data)
    src[src %in% below_threshold] <- "_Other_"
  return(as.factor(src))
}
table(pool_categories(payments$PYMNT_SRC, threshold=150))
```

```

_Other_       A       C       F       G       I       S       U       W 
    278   69144    2092    6690     513   19762    4305   50574 1356858 
```

3. Payment counts and amounts

Timing of payments:

Hide

```
# Creating a date variable and examining the range of dates
payments$DATE <- as.Date(payments$COLLECTION_DT, "%m/%d/%Y")
from <- payments$DATE[which.min(payments$DATE)]
to <- payments$DATE[which.max(payments$DATE)]
paste0("Dates range from ",from," to ",to)
```

```
[1] "Dates range from 2002-07-06 to 2016-11-04"
```

Payments were made between July 6, 2002 and November 4, 2016

Concentration of payments:

Hide

```
# Percentage of total payments made before May 1, 2015
decim <- sum(payments$DATE < as.Date("2015-05-01", "%Y-%m-%d"))/(length(payments$DATE))
perc <- round(decim*100, digits=2)
paste0(perc,"%")
```

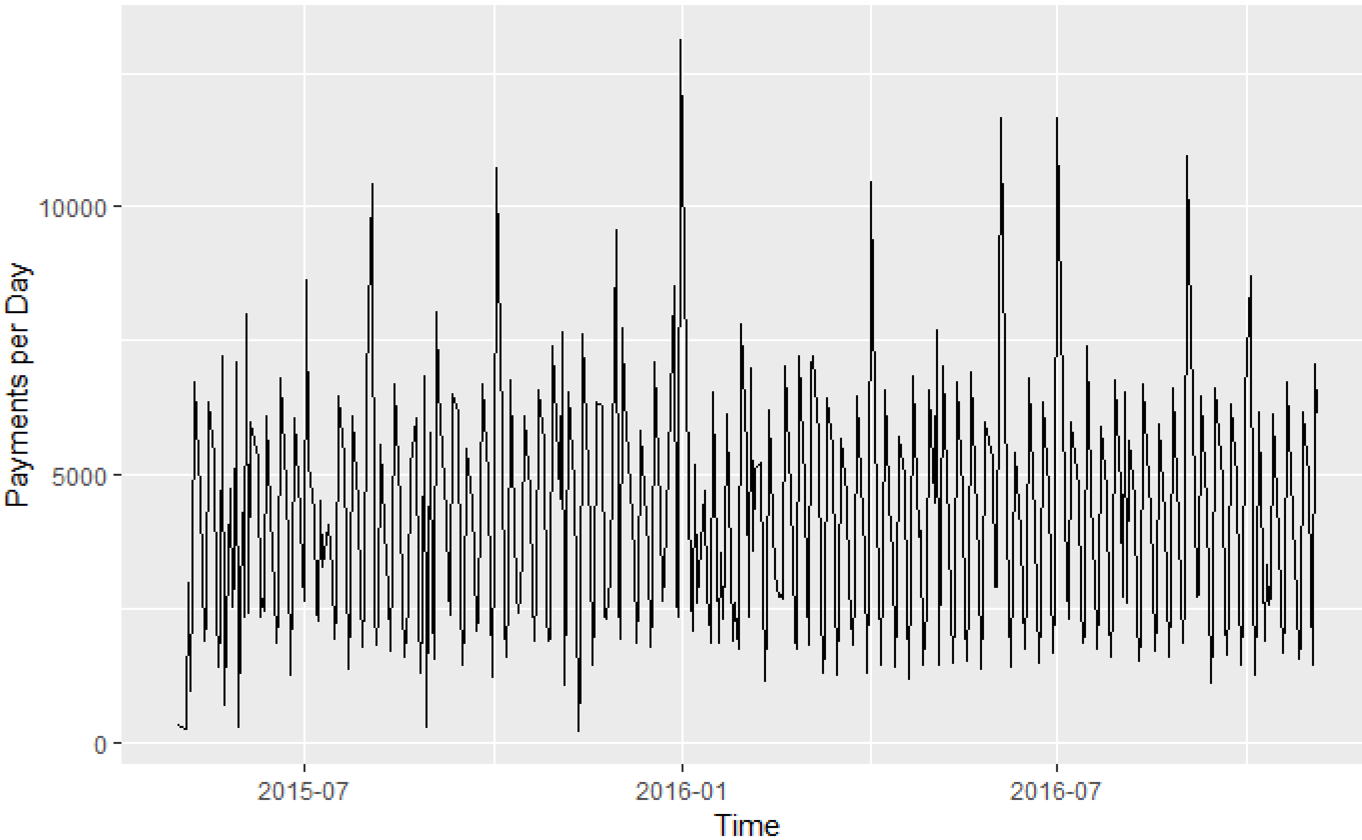
```
[1] "0.38%"
```

Almost all payments were made after May 1, 2015. Only 0.38% of payments occured before then.

Payments per day since May 1,2015:

Hide

```
library(ggplot2)
postMay15 <- as.Date("2015-05-01", "%Y-%m-%d")
payments %>%
  group_by(DATE) %>%
    dplyr::summarize(
      count      = n()
    ) %>%
      filter(DATE >= postMay15) %>%
        ggplot() +
          geom_line(aes(x=DATE, y=count)) +
          labs(y="Payments per Day") +
          labs(x="Time")
```

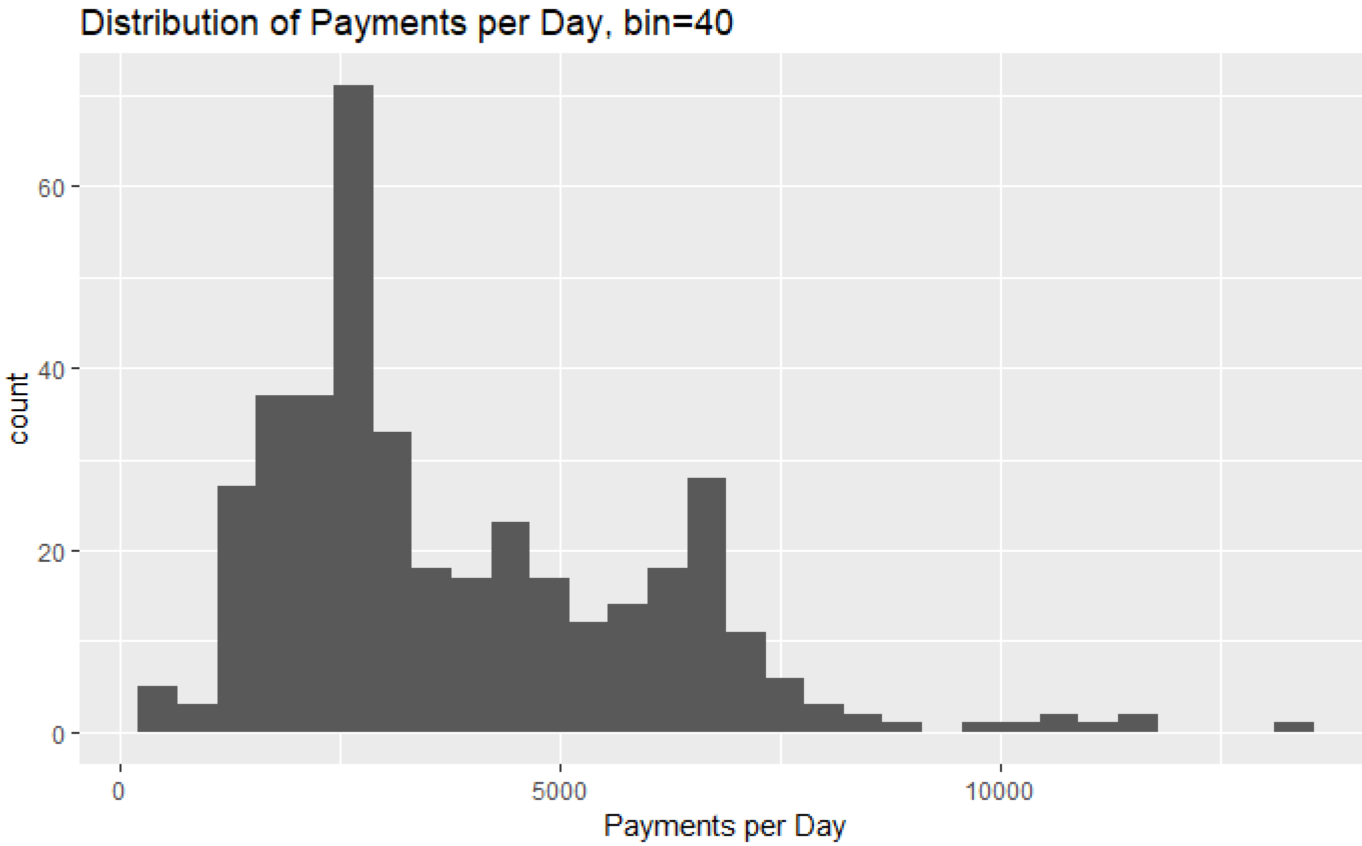


Examining the total number of payments made on each day (from May 1, 2015 through the end of the data) shows repeating instances of days with a very high number of payments.

Distribution of payments per day:

Hide

```
# Distribution of payments per day
payments %>%
  group_by(DATE) %>%
    dplyr::summarize(
      count      = n()
    ) %>%
      filter(DATE >= postMay15) %>%
        ggplot() +
          geom_histogram(aes(count),bins = 30) +
          labs(x="Payments per Day") +
          labs(title = "Distribution of Payments per Day, bin=40")
```

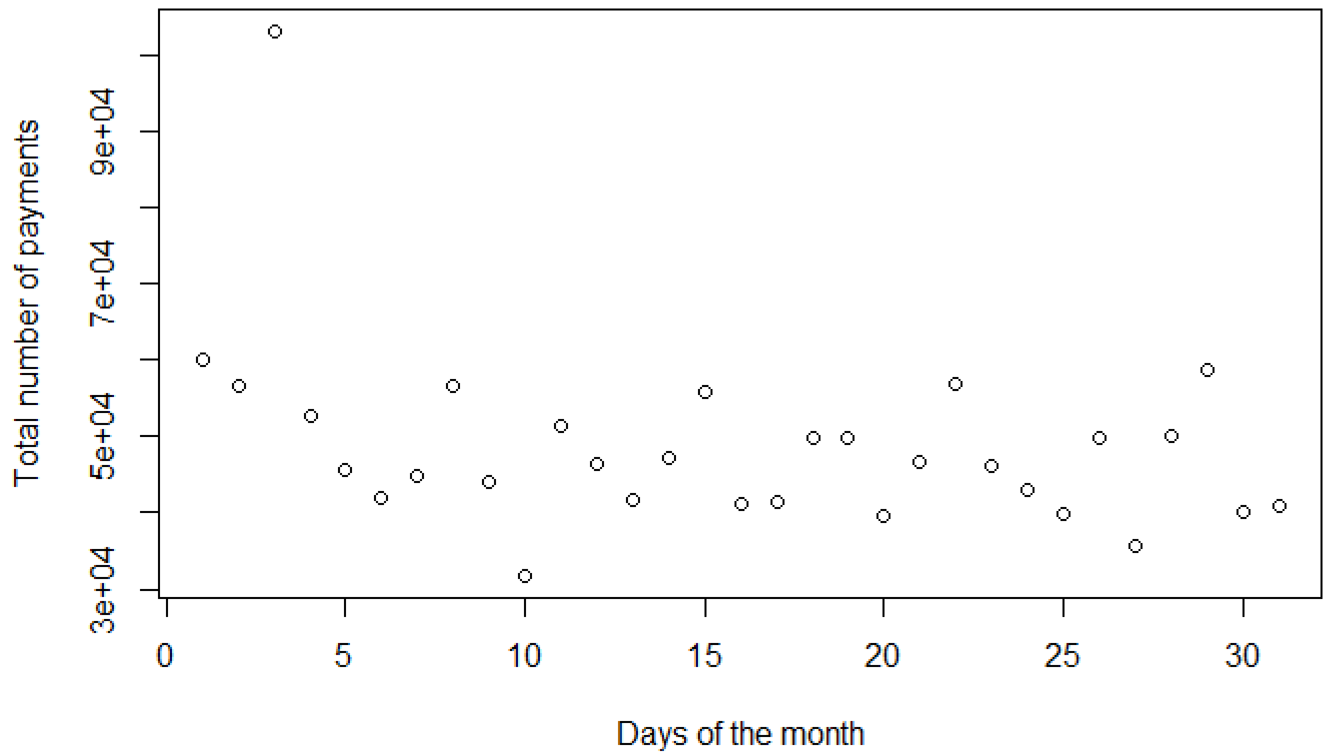


The bimodal distribution suggests that, generally, the number of payments per day is normally distributed (most days it fall around ~2500 payments), but that some days have an usual number of payments

Number of payments per day of month:

Hide

```
# Number of payments by day of the month
y <- as.vector(table(format(payments$DATE, "%d")))
x <- 1:31
plot(x,y, xlab = "Days of the month", ylab = "Total number of payments")
```

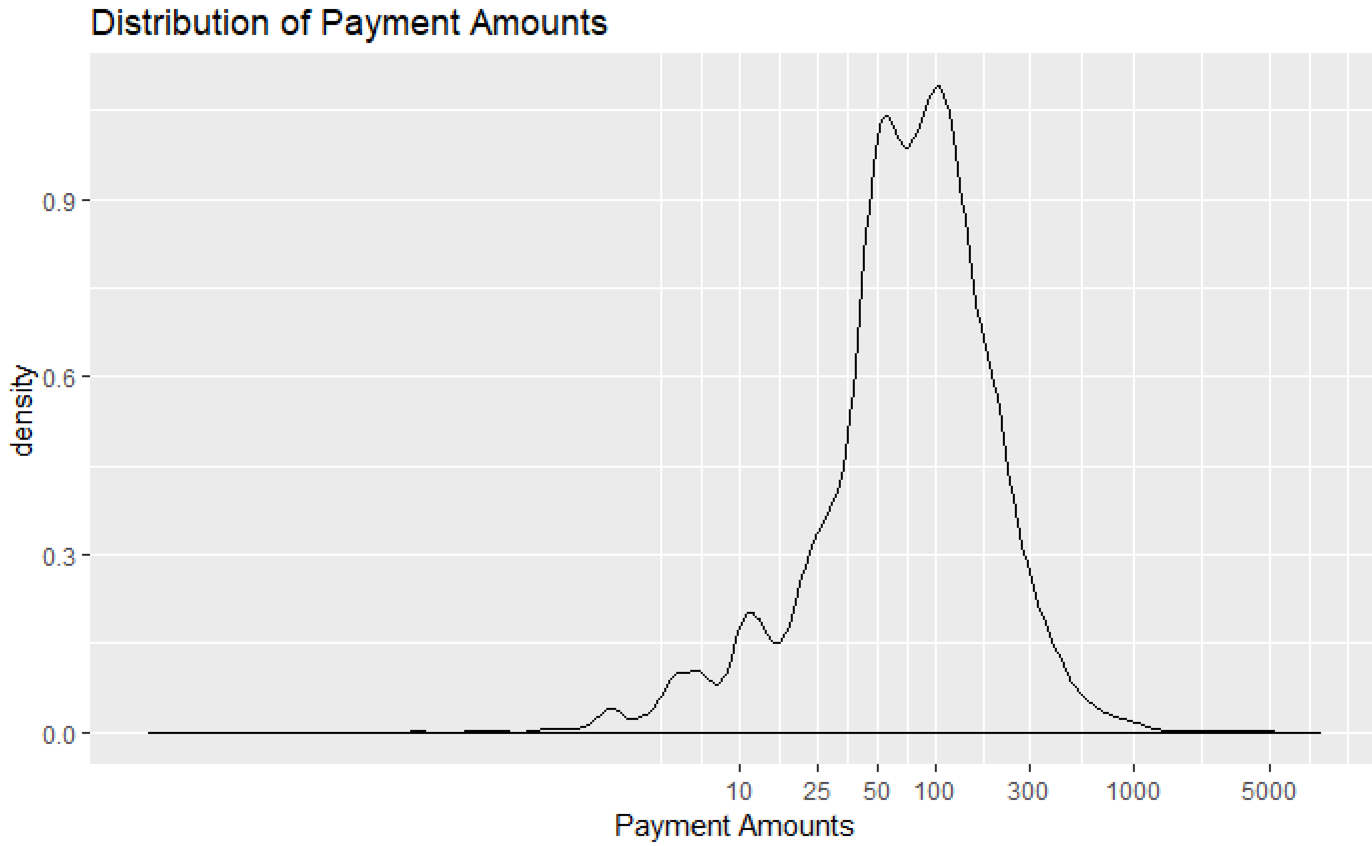


Those payments fall on the 3rd day of the month (likely the due date) when a significant number of absent parents pay child support.

Distribution of payment amounts:

Hide

```
sample1 <- payments[sample(nrow(payments), 10000), ]
ggplot(sample1, aes(x=PYMNT_AMT)) +
  geom_density() +
  scale_x_log10(breaks=c(10,25,50,100,300,1000,5000)) +
  labs(x="Payment Amounts") +
  labs(title = "Distribution of Payment Amounts")
```



The distribution of the payment amounts shows a bumpy distribution curve peaking at $100. The peak suggests that certain payment amounts come more naturally to people's minds than others.

4. Most common parent

Maximum number of cases per parent:

Hide

```
max <- which.max(as.vector(table(cases$AP_ID)))
mostparent <- table(cases$AP_ID)[max]
paste0("Absent parent with ID number ", names(mostparent), " has the most cases (", mostparent,"
 cases)")
```

```
[1] "Absent parent with ID number 1771420 has the most cases (33 cases)"
```

One absent parent has a whopping 33 cases under his or her belt.

Children associates with the 33 cases:

Hide

```
mostcase <- as.integer(filter(cases, AP_ID == names(mostparent))[,"CASE_NUM"])
allkids <- filter(children, CASE_NUM %in% mostcase)
allkids
```

CASE_NUM <int>	ID <int>	SEX... <fctr>	RAC... <fctr>	MARITAL_STS... <fctr>	PRIM_LANG... <fctr>	CITIZENSHIP_CD <fctr>	DATE_(<fctr>
801436400	156329129	F	B	N	E	C	8/6/198
801436400	219324928	F	U	N	E	C	6/12/19
801436400	236322514	F	B	S	E	C	5/8/199
811442925	196340326	F	B	N	E	C	11/14/2
811442925	196340327	M	B	N	E	C	10/10/2
811442925	208356214	M	B	N	E	C	12/30/2
871437126	156338697	F	B	N	E	C	9/8/200
871437126	176347165	F	B	N	E	C	11/3/20
871437126	176347166	M	B	N	E	C	11/3/20
871437126	211338601	M	B	N	E	C	9/8/200

1-10 of 68 rows | 1-8 of 9 columns Previous **1** 2 3 4 5 6 7 Next

Hide

```
#There are 68 children associated with the cases of that parent
```

These 33 cases are associated with 68 different children. That's a lot of child support!

Average of those children:

Hide

```
Age <- (as.Date("03/04/17", "%m/%d/%y") - as.Date(allkids$DATE_OF_BIRTH_DT, "%m/%d/%Y"))/365
MeanAge <- round(mean(Age)[[1]], digits = 1)
paste0("The average age of these children is ", MeanAge, " years")
```

```
[1] "The average age of these children is 17.1 years"
```

The average age of the childrens is 17 years old as of March 4, 2017

Payments made by the parent:

Hide

```
filter(payments, AP_ID == names(mostparent))
```

0 rows

The parent hasn't made a single child support payment (at least not after July 6, 2002)

5. Payments for cases

Relationship between the number of children of each parent and his/her payment frequency and amount:

Hide

```
#Creating a table of total number of payments and total payment amounts for each parent ID
APs <- group_by(payments, AP_ID)
Smry <- dplyr::summarize(APs,
            numpay     = n(),
            totalpay   = sum(PYMNT_AMT)
    )
IDs <- group_by (children, CASE_NUM)
Smry2 <- dplyr::summarize(IDs,
                          numkid = n()     ## number of kids per case number
                          )
Smry3 <- merge(cases, Smry2, by="CASE_NUM")  #Adds kids per case number column to cases
IDs1 <- group_by (Smry3, AP_ID)  #Calculating total number of kids per parent
Smry4 <- dplyr::summarize(IDs1,
                          numkid = sum(numkid)
                          )
newsmry <- merge(x = Smry4, y = Smry[,c("AP_ID","numpay","totalpay")], by = "AP_ID", all.x = TRU
E)
newsmry$numpay[is.na(newsmry[,"numpay"])] <- 0
newsmry$totalpay[is.na(newsmry[,"totalpay"])] <- 0
summary(lm(totalpay ~ numkid, data=newsmry))
```

```
Call:
lm(formula = totalpay ~ numkid, data = newsmry)

Residuals:
   Min     1Q Median     3Q    Max
-35338  -1347   -832   -832 195311

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  316.839     21.240   14.92   <2e-16 ***
numkid       515.016      7.686   67.00   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4667 on 120309 degrees of freedom
Multiple R-squared:  0.03598,   Adjusted R-squared:  0.03597
F-statistic:  4490 on 1 and 120309 DF,  p-value: < 2.2e-16
```

Hide

```
summary(lm(numpay ~ numkid, data=newsmry))
```

```
Call:
lm(formula = numpay ~ numkid, data = newsmry)

Residuals:
   Min     1Q Median     3Q    Max 
-468.1  -11.5   -4.6   -4.6 9006.4 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -2.33317    0.24911  -9.366   <2e-16 ***
numkid       6.91809    0.09015  76.742   <2e-16 ***
---
Signif. codes:  0 □***□ 0.001 □**□ 0.01 □*□ 0.05 □.□ 0.1 □ □ 1

Residual standard error: 54.73 on 120309 degrees of freedom
Multiple R-squared:  0.04667,	Adjusted R-squared:  0.04666 
F-statistic:  5889 on 1 and 120309 DF,  p-value: < 2.2e-16
```

Parents responsible for more children are more likely to make either a larger number of payments or a larger total payment amount over this period. Running two separate linear regressions, we see that the number of children is predictive of both total number of payments and total payment amount ($p < 0.05$).

Relationship between the age of the children and the cumulative payment amounts of the parents:

Hide

```
IDs1 <- group_by (children, CASE_NUM)
TotAgeCase <- dplyr::summarize(IDs1,
                          totage = as.numeric(sum((as.Date("03/04/17", "%m/%d/%y")-as.Date(DATE_
OF_BIRTH_DT, "%m/%d/%Y"))/365)[[1]])
                          )
TotAgeCase1 <- merge(cases, TotAgeCase, by="CASE_NUM")
APs1 <- group_by(TotAgeCase1, AP_ID)
TotAgeParent <- dplyr::summarize(APs1,
                                 totalage = sum(totage)
                                 )
FinalSmry1 <- merge(newsmry, TotAgeParent, by="AP_ID")
head(FinalSmry1)
```

	AP_ID <int>	numkid <int>	numpay <dbl>	totalpay <dbl>	totalage <dbl>
1	1718626	1	10	5175.32	8.243836
2	1718627	2	0	0.00	11.775342
3	1718628	1	36	9360.00	13.657534
4	1718629	7	5	2756.20	103.953425
5	1718630	1	15	2186.10	10.449315
6	1718632	3	0	0.00	34.649315

6 rows

Hide

```
FinalSmry1$avgage <- FinalSmry1$totalage / FinalSmry1$numkid
summary(lm(totalpay ~ avgage, data=FinalSmry1))
```

```

Call:
lm(formula = totalpay ~ avgage, data = FinalSmry1)

Residuals:
   Min     1Q Median     3Q    Max 
 -2947  -1684  -1028   -406 206953 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 2977.800     32.732   90.98   <2e-16 ***
avgage       -76.263      1.457  -52.34   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4702 on 119937 degrees of freedom
  (372 observations deleted due to missingness)
Multiple R-squared:  0.02233,   Adjusted R-squared:  0.02232 
F-statistic:  2739 on 1 and 119937 DF,  p-value: < 2.2e-16
```

The negative coefficient (with p<0.05) estimate suggests that parents with older children of on average pay a lower total amount, which makes sense; as children get older they rely less and less on financial support.

Relationship between parent location and cumulative payments made by parents:

Hide

```
FinalSmry2 <- subset(merge(newsmry, parents, by="AP_ID"), numpay != 0)
summary(lm(totalpay ~ AP_ADDR_ZIP, data=FinalSmry2))
```

```
Call:
lm(formula = totalpay ~ AP_ADDR_ZIP, data = FinalSmry2)

Residuals:
   Min     1Q Median     3Q    Max 
 -7274  -4662  -2373   1409 202481 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    6608.10    5840.38   1.131    0.258
AP_ADDR_ZIP01  -585.47    5840.72  -0.100    0.920
AP_ADDR_ZIP02   671.19    5849.84   0.115    0.909
AP_ADDR_ZIP03    98.42    5841.17   0.017    0.987
AP_ADDR_ZIP04 -1855.27    5842.23  -0.318    0.751
AP_ADDR_ZIPna -3497.41    6243.63  -0.560    0.575

Residual standard error: 8260 on 28030 degrees of freedom
Multiple R-squared:  0.004941,  Adjusted R-squared:  0.004763 
F-statistic: 27.84 on 5 and 28030 DF,  p-value: < 2.2e-16
```

The parent location (AP_ADDR_ZIP) is indicative of the total amount of payments made by the absent parent (F-test: $p<0.05$). Parents outside of cities pay the most (ZIP02 and ZIP03).

Relationship between parent attributes and cumulative payments:

Checking what levels correspond to a missing value and fixing columns

Hide

```
Smry5 <- merge(FinalSmry1, parents, by="AP_ID")
levels(Smry5$AP_ADDR_ZIP) # "na"
```

```
[1] "00" "01" "02" "03" "04" "na"
```

Hide

```
levels(Smry5$AP_DECEASED_IND) #" "
```

```
[1] " " "N" "Y"
```

Hide

```
levels(Smry5$MARITAL_STS_CD) #" "
```

```
[1] " " "D" "M" "N" "S" "W"
```

Hide

```
levels(Smry5$SEX_CD) #"U"
```

```
[1] "F" "M" "U"
```

Hide

```
levels(Smry5$RACE_CD) #"U"
```

```
[1] "A" "B" "C" "H" "N" "P" "U"
```

Hide

```
levels(Smry5$PRIM_LANG_CD) #""
```

```
 [1] ""  "A" "E" "F" "G" "H" "I" "K" "L" "O" "P" "R" "S" "V" "X" "Z"
```

Hide

```
levels(Smry5$CITIZENSHIP_CD) #""
```

```
[1] ""  "C" "I" "L" "R"
```

Hide

```
#Fixing columns
Unknowns <- function(data, string, sub) {
  data1 <- as.character(data)
  data1[data1 == string] <- sub
  data1 <- as.factor(data1)
return(data1)
  }
Smry5$AP_ADDR_ZIP <- Unknowns(Smry5$AP_ADDR_ZIP,"na","U")
Smry5$AP_DECEASED_IND <- Unknowns(Smry5$AP_DECEASED_IND," ","U")
Smry5$MARITAL_STS_CD <- Unknowns(Smry5$MARITAL_STS_CD," ","U")
Smry5$PRIM_LANG_CD <- Unknowns(Smry5$PRIM_LANG_CD,"","U")
Smry5$CITIZENSHIP_CD <- Unknowns(Smry5$CITIZENSHIP_CD,"","U")
any(!is.na(Smry5$AP_CUR_INCAR_IND))   # All in this category are NAs - I will not include it
```

```
[1] FALSE
```

Checking what levels correspond to a missing value and fixing columns

Hide

```
#Running regression
summary(lm(totalpay ~ numpay + numkid + AP_ADDR_ZIP + AP_DECEASED_IND + AP_APPROX_AGE + MARITAL_
STS_CD + SEX_CD + RACE_CD + PRIM_LANG_CD + CITIZENSHIP_CD, data=Smry5))
```

```
Call:
lm(formula = totalpay ~ numpay + numkid + AP_ADDR_ZIP + AP_DECEASED_IND +
    AP_APPROX_AGE + MARITAL_STS_CD + SEX_CD + RACE_CD + PRIM_LANG_CD +
    CITIZENSHIP_CD, data = Smry5)

Residuals:
    Min      1Q  Median      3Q     Max
-344610    -921    -461      44   92486

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       1836.3281   452.7140   4.056 4.99e-05 ***
numpay              60.0008     0.1692 354.670  < 2e-16 ***
numkid              46.0490     5.5513   8.295  < 2e-16 ***
AP_ADDR_ZIP01      522.9277    33.9972  15.381  < 2e-16 ***
AP_ADDR_ZIP02      727.9900    68.1260  10.686  < 2e-16 ***
AP_ADDR_ZIP03      844.5573    37.9451  22.257  < 2e-16 ***
AP_ADDR_ZIP04      260.9038    40.9279   6.375 1.84e-10 ***
AP_ADDR_ZIPU       -29.6344    49.3976  -0.600  0.54856
AP_DECEASED_INDU  -297.2300    38.9853  -7.624 2.47e-14 ***
AP_DECEASED_INDY  -720.3165    58.6791 -12.276  < 2e-16 ***
AP_APPROX_AGE        4.2111     0.8522   4.941 7.76e-07 ***
MARITAL_STS_CDM   -903.8244   113.5663  -7.959 1.76e-15 ***
MARITAL_STS_CDN  -1638.9744    78.5922 -20.854  < 2e-16 ***
MARITAL_STS_CDS   -536.0736    94.4498  -5.676 1.38e-08 ***
MARITAL_STS_CDU  -2085.0761    77.9683 -26.743  < 2e-16 ***
MARITAL_STS_CDW  -1363.1047   420.6195  -3.241  0.00119 **
SEX_CDM            447.4684    28.2020  15.867  < 2e-16 ***
SEX_CDU            457.3386    68.6861   6.658 2.78e-11 ***
RACE_CDB          -310.4203   257.2144  -1.207  0.22749
RACE_CDC          -460.8723   259.2235  -1.778  0.07542 .
RACE_CDH          -694.0202   270.4439  -2.566  0.01028 *
RACE_CDN          -808.6652   352.0193  -2.297  0.02161 *
RACE_CDP          -546.2113   652.9457  -0.837  0.40286
RACE_CDU          -798.7418   258.1039  -3.095  0.00197 **
PRIM_LANG_CDE      314.6882   362.4425   0.868  0.38526
PRIM_LANG_CDF     1555.7184   856.0029   1.817  0.06916 .
PRIM_LANG_CDG     -252.5291  2275.0408  -0.111  0.91162
PRIM_LANG_CDH      691.0059   614.2289   1.125  0.26059
PRIM_LANG_CDI      156.5237   493.2805   0.317  0.75101
PRIM_LANG_CDK      241.3726  3196.8832   0.076  0.93982
PRIM_LANG_CDL     -150.5956  1119.0196  -0.135  0.89295
PRIM_LANG_CDO      239.4560   456.3419   0.525  0.59977
PRIM_LANG_CDP     -140.6431  1465.9652  -0.096  0.92357
PRIM_LANG_CDR     1210.3061   573.5925   2.110  0.03486 *
PRIM_LANG_CDS      461.4325   464.1909   0.994  0.32020
PRIM_LANG_CDU      357.5145   363.9238   0.982  0.32591
PRIM_LANG_CDV      -46.8192  1630.3357  -0.029  0.97709
PRIM_LANG_CDX      379.6887  1028.0071   0.369  0.71187
PRIM_LANG_CDZ     -310.0531  2278.2814  -0.136  0.89175
CITIZENSHIP_CDI    291.5127   176.7113   1.650  0.09902 .
CITIZENSHIP_CDL    360.4877   171.8307   2.098  0.03591 *
```

```
CITIZENSHIP_CDR   2831.4025  1296.9582   2.183  0.02903 *  
CITIZENSHIP_CDU    416.8293    26.1919  15.914  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3176 on 120268 degrees of freedom
Multiple R-squared:  0.5536,	Adjusted R-squared:  0.5534 
F-statistic:  3551 on 42 and 120268 DF,  p-value: < 2.2e-16
```

Running a multivariate regression indicates that the following factors are predictive of the total amount paid in child support by an absent parent: Number of payments made, number of kids, location, marital status, sex, and citizenship

Note that although some race variables have p-values below 0.05, race may still not be predictive. Adjusting for multiple tests using Bonferoni correction suggests it is not.

6. Payment consistency

An important aspect of payments is the consistency of the payments over time.

Relationship between volatility of payments and avg. size of payments:

Hide

```
DF1 <- payments %>%
  group_by(AP_ID) %>%
    summarize(
      avgpay = mean(PYMNT_AMT),
      sdpay = sd(PYMNT_AMT)
    )
inconsistent_payers <- subset(DF1, sdpay != 0)
summary(lm(sdpay ~ avgpay, data=inconsistent_payers))
```

```
Call:
lm(formula = sdpay ~ avgpay, data = inconsistent_payers)

Residuals:
    Min      1Q  Median      3Q     Max 
-4925.8   -55.9    -7.4    27.7  5949.9 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -34.285785   1.809235  -18.95   <2e-16 ***
avgpay        0.976397   0.004987  195.78   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 229.3 on 20799 degrees of freedom
Multiple R-squared:  0.6482,	Adjusted R-squared:  0.6482 
F-statistic: 3.833e+04 on 1 and 20799 DF,  p-value: < 2.2e-16
```

A steady income stream is, for many, preferable to a highly volatile, unpredictable payment schedule, even if the latter has a higher average. Among parents who make inconsistent payments, those who make larger daily payments also make more volatile payments ($p<0.05$).

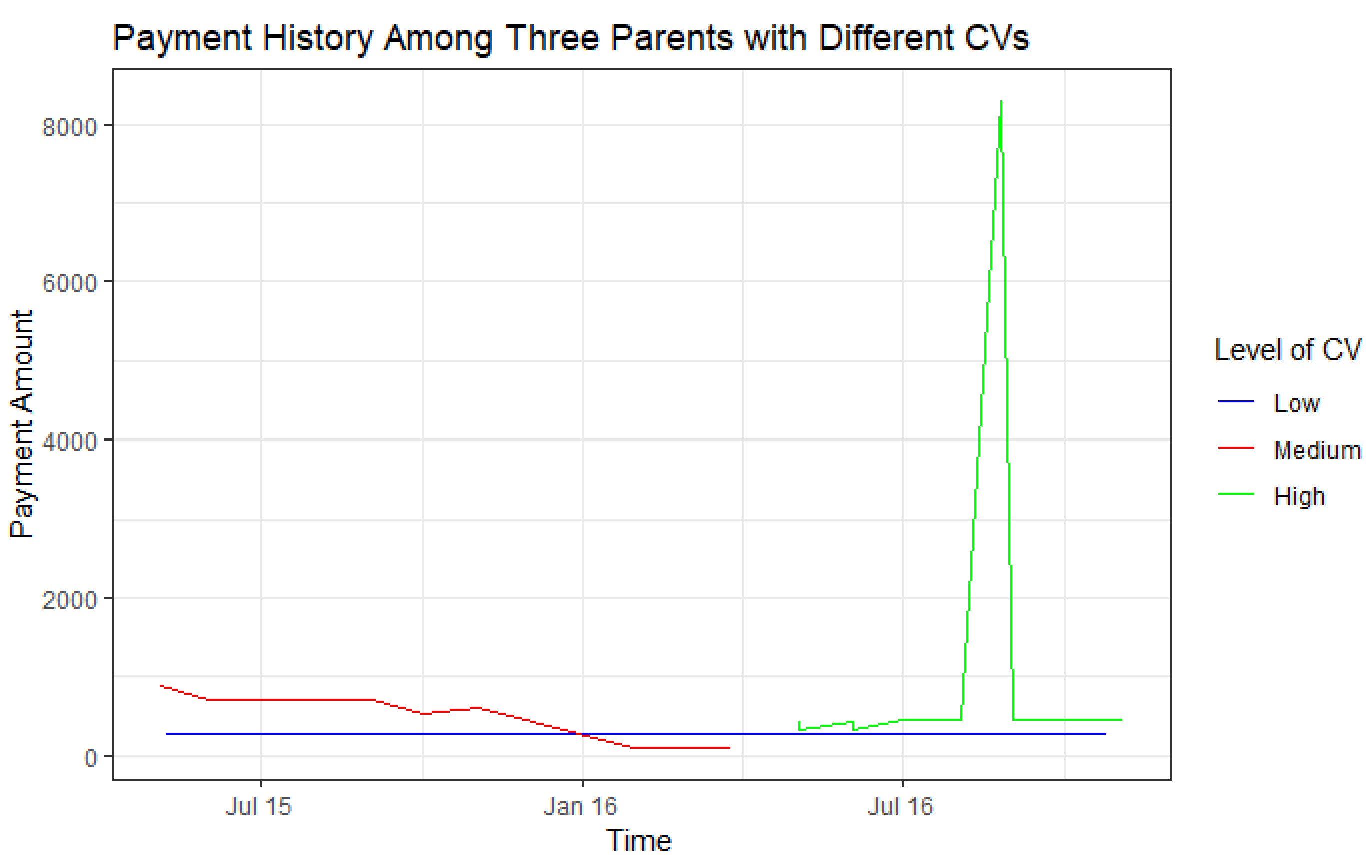
Time plot of three parent examples with low, high, and medium coefficient variation (CV):

Hide

```
DF2 <- payments %>%
  group_by(AP_ID) %>%
    summarize(
      numpay = n(),
      avgpay = mean(PYMNT_AMT),
      sdpay = sd(PYMNT_AMT),
      CV = sdpay/avgpay
    ) %>%
  filter(numpay > 3)
percs <- quantile(DF2$CV, c(0.1,0.45, 0.55, 0.9), na.rm = TRUE)
IDLow <- DF2$AP_ID[which(DF2$CV <= percs[[1]])[1]]
IDMedium <- DF2$AP_ID[which(percs[[2]] <= DF2$CV | DF2$CV <= percs[[3]])[1]]
IDHigh <- DF2$AP_ID[which(DF2$CV > percs[[4]])[1]]
AllIDs <- c(IDLow, IDMedium, IDHigh)
DF3 <- payments %>%
    filter(AP_ID %in% AllIDs) %>%
      arrange(AP_ID, DATE) %>%
        group_by(AP_ID)

DF3$AP_ID <- factor(DF3$AP_ID, levels = c(IDLow, IDMedium, IDHigh))

ggplot(data=DF3, aes(x=DATE, y=PYMNT_AMT, color=AP_ID)) +
  geom_line() +
  theme_bw() +
  scale_x_date(date_labels = "%b %y") +
  scale_color_manual(labels = c("Low", "Medium", "High"), values = c("blue", "red", "green")) +
  labs(x="Time", y="Payment Amount", title="Payment History Among Three Parents with Different C
Vs", color="Level of CV")
```



Relationship between CV and cumulative payments:

Hide

```
#Only includes those parents who have made more than one payments, since those who've only paid
 once automatically have a CV of 0, distorting results
DF4 <- payments %>%
  group_by(AP_ID) %>%
    summarize(
      numpay = n(),
      totpay = sum(PYMNT_AMT),
      avgpay = mean(PYMNT_AMT),
      sdpay = sd(PYMNT_AMT),
      CV = sdpay/avgpay
    ) %>%
  filter(numpay > 1)
summary(lm(CV ~ totpay, data=DF4))
```

```
Call:
lm(formula = CV ~ totpay, data = DF4)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.6885 -0.4870 -0.2162  0.1745  9.5334 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 4.831e-01  5.570e-03   86.72   <2e-16 ***
totpay      9.306e-06  5.335e-07   17.44   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7353 on 27230 degrees of freedom
Multiple R-squared:  0.01105,	Adjusted R-squared:  0.01101 
F-statistic: 304.2 on 1 and 27230 DF,  p-value: < 2.2e-16
```

On average, the more a parent pays over a given time period, the more volatile are the payments in terms of CV (p-value < 0.05), which is consistent with the first observation in issue 6.

Relationship between volatility of payments and parent attributes:

Hide

```
#Only includes those parents who have made more than one payments, since those who've only paid
 once automatically have a CV of 0, distorting results
DF5 <- merge(DF4, parents, by="AP_ID")
DF5$AP_ADDR_ZIP <- Unknowns(DF5$AP_ADDR_ZIP,"na","U")
DF5$AP_DECEASED_IND <- Unknowns(DF5$AP_DECEASED_IND," ","U")
DF5$MARITAL_STS_CD <- Unknowns(DF5$MARITAL_STS_CD," ","U")
DF5$PRIM_LANG_CD <- Unknowns(DF5$PRIM_LANG_CD,"","U")
DF5$CITIZENSHIP_CD <- Unknowns(DF5$CITIZENSHIP_CD,"","U")
summary(lm(CV ~ AP_ADDR_ZIP + AP_DECEASED_IND + AP_APPROX_AGE + MARITAL_STS_CD + SEX_CD + RACE_C
D + PRIM_LANG_CD + CITIZENSHIP_CD, data=DF5))
```

```
Call:
lm(formula = CV ~ AP_ADDR_ZIP + AP_DECEASED_IND + AP_APPROX_AGE + 
    MARITAL_STS_CD + SEX_CD + RACE_CD + PRIM_LANG_CD + CITIZENSHIP_CD, 
    data = DF5)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.0325 -0.4729 -0.2215  0.1677  9.5726 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)       0.0634809  0.7669052   0.083 0.934031    
AP_ADDR_ZIP01     0.4946526  0.7359061   0.672 0.501482    
AP_ADDR_ZIP02     0.5118886  0.7365044   0.695 0.487046    
AP_ADDR_ZIP03     0.5132366  0.7358945   0.697 0.485538    
AP_ADDR_ZIP04     0.5768561  0.7359898   0.784 0.433174    
AP_ADDR_ZIPU      0.7024773  0.7723243   0.910 0.363061    
AP_DECEASED_INDU -0.0503187  0.0134781  -3.733 0.000189 ***
AP_DECEASED_INDY -0.2150682  0.0631801  -3.404 0.000665 ***
AP_APPROX_AGE    -0.0000529  0.0003343  -0.158 0.874258    
MARITAL_STS_CDM   0.0858925  0.0369827   2.323 0.020213 *  
MARITAL_STS_CDN   0.0931823  0.0254178   3.666 0.000247 ***
MARITAL_STS_CDS   0.0257311  0.0313019   0.822 0.411068    
MARITAL_STS_CDU  -0.0139892  0.0254208  -0.550 0.582115    
MARITAL_STS_CDW  -0.0141463  0.1521292  -0.093 0.925913    
SEX_CDM          -0.0942076  0.0219036  -4.301 1.71e-05 ***
SEX_CDU          -0.1275176  0.1304854  -0.977 0.328452    
RACE_CDB          0.0194582  0.1192366   0.163 0.870370    
RACE_CDC          0.0211978  0.1205739   0.176 0.860447    
RACE_CDH         -0.0181761  0.1276163  -0.142 0.886743    
RACE_CDN         -0.0257232  0.1891158  -0.136 0.891808    
RACE_CDP         -0.5492011  0.5339941  -1.028 0.303734    
RACE_CDU          0.0172586  0.1199450   0.144 0.885590    
PRIM_LANG_CDE     0.0676234  0.1797486   0.376 0.706763    
PRIM_LANG_CDF     0.3463095  0.3318477   1.044 0.296689    
PRIM_LANG_CDH    -0.1089874  0.3162555  -0.345 0.730384    
PRIM_LANG_CDI     0.4387245  0.2362553   1.857 0.063323 .  
PRIM_LANG_CDL    -0.2534319  0.5502741  -0.461 0.645121    
PRIM_LANG_CDO    -0.0359229  0.2206886  -0.163 0.870696    
PRIM_LANG_CDR     0.1445807  0.2720152   0.532 0.595065    
PRIM_LANG_CDS     0.0185242  0.2221756   0.083 0.933553    
PRIM_LANG_CDU     0.0945129  0.1805411   0.523 0.600632    
PRIM_LANG_CDX    -0.0181652  0.4105019  -0.044 0.964704    
CITIZENSHIP_CDI  -0.1167223  0.0850350  -1.373 0.169876    
CITIZENSHIP_CDL   0.0126888  0.0723853   0.175 0.860849    
CITIZENSHIP_CDR   0.0650957  0.5208843   0.125 0.900547    
CITIZENSHIP_CDU  -0.1206883  0.0133243  -9.058  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7354 on 27196 degrees of freedom
```

```
Multiple R-squared:  0.0122,    Adjusted R-squared:  0.01093
F-statistic: 9.596 on 35 and 27196 DF,  p-value: < 2.2e-16
```

The following regression shows that certain parent attributes are actually indicative of more consistent payments. The volatility the can be expected to be lower on average when: 1. The parent is either deceased or living status is unknown 2. The parent's marital status is N (p-value for marital status M is too close to 0.05 given number of parameters to be considered significant) 3. The parent is male 4. The parent's citizenship is unknown