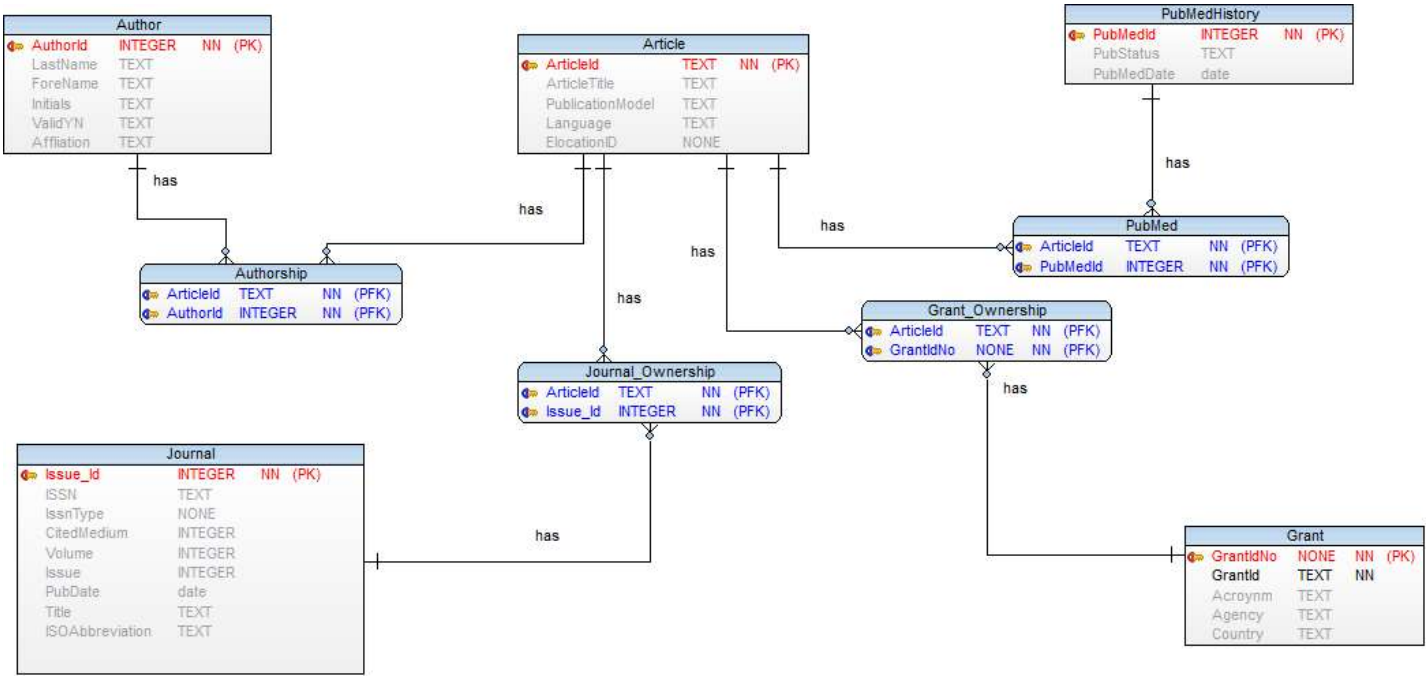
Assignment 10

Part 1

Create a normalized relational schema that contains minimally the following entities: Article, Journal, Author, History. Use the XML document to determine the appropriate attributes (fields/columns) for the entities (tables). While there may be other types of publications in the XML, you only need to deal with articles in journals. Create appropriate primary and foreign keys. Where necessary, add surrogate keys. Include an image of an ERD showing your model in your R Notebook.



ERD Diagram Design

```
library(RSQLite)
```

```
## Warning: package 'RSQLite' was built under R version 3.6.3
```

```
con <- dbConnect(RSQLite::SQLite(), "PubMedRDBM.db")

dbListTables(con)
```

```
##  [1] "Article"           "Author"            "Authorship"
##  [4] "FactTable"         "Grant"             "Grant_Ownership"
##  [7] "Journal"           "Journal_Ownership" "PubMed"
## [10] "PubMedHistory"
```

2. Realize the relational schema in SQLite (place the CREATE TABLE statements into SQL chunks in your R Notebook).

dbExecute(con, “PRAGMA foreign_keys = OFF;”)

dbExecute(con, “CREATE TABLE Article (ArticleId TEXT NOT NULL, ArticleTitle TEXT, PublicationModel TEXT, Language TEXT, ElocationID TEXT, CONSTRAINT PK_Article PRIMARY KEY (ArticleId));”)

dbExecute(con, “CREATE TABLE Authorship (ArticleId TEXT NOT NULL, AuthorId INTEGER NOT NULL, CONSTRAINT PK_Authorship PRIMARY KEY (ArticleId,AuthorId), CONSTRAINT has FOREIGN KEY (ArticleId) REFERENCES Article (ArticleId), CONSTRAINT has FOREIGN KEY (AuthorId) REFERENCES Author (AuthorId));”)

dbExecute(con, " CREATE TABLE Author (AuthorId INTEGER NOT NULL, LastName TEXT, ForeName TEXT, Initials TEXT, ValidYN TEXT, Affiliation TEXT, CONSTRAINT PK_Author PRIMARY KEY (AuthorId)); ")

dbExecute(con, “CREATE TABLE Journal (Issue_Id INTEGER NOT NULL, ISSN TEXT, CitedMedium INTEGER, Volume INTEGER, Issue INTEGER, PubDate date, Title TEXT, ISOAbbreviation TEXT, CONSTRAINT PK_Journal PRIMARY KEY (Issue_Id));”)

dbExecute(con,"

CREATE TABLE Journal_Ownership (ArticleId TEXT NOT NULL, Issue_Id INTEGER NOT NULL, CONSTRAINT PK_Journal_Ownership PRIMARY KEY (ArticleId,Issue_Id), CONSTRAINT has FOREIGN KEY (ArticleId) REFERENCES Article (ArticleId), CONSTRAINT has FOREIGN KEY (Issue_Id) REFERENCES Journal (Issue_Id)); ")

dbExecute(con," CREATE TABLE Grant (GrantIdNo INTEGER NOT NULL, GrantId TEXT NOT NULL, Acronym TEXT, Agency TEXT, Country TEXT, CONSTRAINT PK_Grant PRIMARY KEY (GrantIdNo)); ")

dbExecute(con, “CREATE TABLE Grant_Ownership (ArticleId TEXT NOT NULL, GrantIdNo NONE NOT NULL, CONSTRAINT PK_Grant_Ownership PRIMARY KEY (ArticleId,GrantIdNo), CONSTRAINT has FOREIGN KEY (ArticleId) REFERENCES Article (ArticleId), CONSTRAINT has FOREIGN KEY (GrantIdNo) REFERENCES Grant (GrantIdNo));”)

dbExecute(con, “CREATE TABLE PubMedHistory (PubMedId INTEGER NOT NULL, PubStatus TEXT, PubMedDate date, CONSTRAINT PK_PubMedHistory PRIMARY KEY (PubMedId));”)

dbExecute(con,“CREATE TABLE PubMed (ArticleId TEXT NOT NULL, PubMedId INTEGER NOT NULL, CONSTRAINT PK_PubMed PRIMARY KEY (ArticleId,PubMedId), CONSTRAINT has FOREIGN KEY (ArticleId) REFERENCES Article (ArticleId), CONSTRAINT has FOREIGN KEY (PubMedId) REFERENCES PubMedHistory (PubMedId));”)

Schemas

```
dbGetQuery(con, "pragma table_info('Author')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	AuthorId	INTEGER	1	*NA*	1
1	LastName	TEXT	0	*NA*	0
2	ForeName	TEXT	0	*NA*	0
3	Initials	TEXT	0	*NA*	0
4	ValidYN	TEXT	0	*NA*	0
5	Affiliation	TEXT	0	*NA*	0

6 rows

```
dbGetQuery(con, "pragma table_info('Article')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	ArticleId	TEXT	1	*NA*	1
1	ArticleTitle	TEXT	0	*NA*	0
2	PublicationModel	TEXT	0	*NA*	0
3	Language	TEXT	0	*NA*	0
4	ElocationID	TEXT	0	*NA*	0

5 rows

```
dbGetQuery(con, "pragma table_info('PubMedHistory')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	PubMedId	INTEGER	1	*NA*	1
1	PubStatus	TEXT	0	*NA*	0
2	PubMedDate	date	0	*NA*	0

3 rows

```
dbGetQuery(con, "pragma table_info('Journal')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	Issue_Id	INTEGER	0	*NA*	0
1	ISSN	TEXT	0	*NA*	0
2	IssnType	TEXT	0	*NA*	0
3	CitedMedium	TEXT	0	*NA*	0
4	Volume	TEXT	0	*NA*	0
5	Issue	TEXT	0	*NA*	0
6	PubDate	TEXT	0	*NA*	0
7	Title	TEXT	0	*NA*	0
8	ISOAbbreviation	TEXT	0	*NA*	0

9 rows

```
dbGetQuery(con, "pragma table_info('Grant')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	GrantIdNo	INTEGER	1	*NA*	1
1	GrantId	TEXT	1	*NA*	0
2	Acronym	TEXT	0	*NA*	0
3	Agency	TEXT	0	*NA*	0
4	Country	TEXT	0	*NA*	0

5 rows

```
dbGetQuery(con, "pragma table_info('Authorship')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	ArticleId	TEXT	1	*NA*	1
1	AuthorId	INTEGER	1	*NA*	2

2 rows

```
dbGetQuery(con, "pragma table_info('Grant_Ownership')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	ArticleId	TEXT	1	*NA*	1
1	GrantIdNo	NONE	1	*NA*	2

2 rows

```
dbGetQuery(con, "pragma table_info('Journal_Ownership')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	ArticleId	TEXT	1	*NA*	1
1	Issue_Id	INTEGER	1	*NA*	2

2 rows

```
dbGetQuery(con, "pragma table_info('PubMed')")
```

cid <int>	name <chr>	type <chr>	notnull <int>	dflt_value <lgl>	pk <int>
0	ArticleId	TEXT	1	*NA*	1
1	PubMedId	INTEGER	1	*NA*	2

2 rows

Extract and transform the data from the XML and then load into the appropriate tables in the database. You cannot use xmlToDataFrame but instead must parse the XML node by node using a combination of node-by-node tree traversal and XPath. It is not feasible to use XPath to extract all journals, then all authors, etc. as some are missing and won't match up. You will need to iterate through the top-level nodes.

Article

```
ArticleTitle<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["ArticleTitle"]]))

PublishModel<-xmlSApply(root_,function(x)xmlAttrs(x[["MedlineCitation"]][["Article"]]))

Language<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Language"]]))

ELocationID<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][['ELocationID']]))

df.Article<-tibble::rowid_to_column(data.frame("ArticleTitle"=unique(ArticleTitle)), "ArticleId")

df.Article$PublicationModel<-PublishModel

df.Article$Language<-Language

df.Article$ELocationID<-ELocationID

dbWriteTable(con,"Article",df.Article, append=TRUE)
```

Journal

```
IssnType<-xmlSApply(root_,function(x)xmlAttrs(x[["MedlineCitation"]][["Article"]][["Journal"]][["ISSN"]]))

ISSN<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Journal"]][["ISSN"]]))

Citedmedium<-xmlSApply(root_,function(x)xmlAttrs(x[["MedlineCitation"]][["Article"]][["Journal"]][["JournalIssue"]]))

Volume<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Journal"]][["JournalIssue"]]
[["Volume"]]))

Issue<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Journal"]][["JournalIssue"]][["Issue"]]))

PubDate<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Journal"]][["JournalIssue"]]
[["PubDate"]]))

Title<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Journal"]][["Title"]]))

ISOAbbreviation<-xmlSApply(root_,function(x)xmlValue(x[["MedlineCitation"]][["Article"]][["Journal"]]
[["ISOAbbreviation"]]))

df.Journal<-data.frame("ISSN"=ISSN,"IssnType"=IssnType,"CitedMedium"=Citedmedium,
"Volume"=Volume,"Issue"=Issue,"PubDate"=PubDate,"Title"=Title,"ISOAbbreviation"=ISOAbbreviation)
df.Journal$ArticleId<-seq(1:19)

df.Journal<-tibble::rowid_to_column(data.frame(df.Journal), "Issue_Id")
```

dbWriteTable(con,“Journal”,df.Journal[,-c(10)], append=TRUE)
dbWriteTable(con,“Journal_Ownership”,df.Journal[,c(10,1)], append=TRUE)

Author

LastName<-c() ForeName<-c() Initials<-c() ValidYN<-c() Affiliation<-c() ArticleId<-c()

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]])))) { LastName<-c(LastName,xmlValue(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]][[j]][[“LastName”]])) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]])))) { Initials<-c(Initials,xmlValue(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]][[j]][[“Initials”]])) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]])))) { ValidYN<-c(ValidYN,xmlAttrs(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]][[j]])) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]])))) { ForeName<-c(ForeName,xmlValue(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]][[j]][[“ForeName”]])) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]])))) { Affiliation<-c(Affiliation,xmlValue(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]][[j]][[“Affiliation”]])) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“MedlineCitation”]][[“Article”]][[“AuthorList”]])))) { ArticleId<-c(ArticleId,i) } }

df.Author<-
data.frame(“LastName”=LastName,“ForeName”=ForeName,“Initials”=Initials,“ValidYN”=ValidYN,“Affiliation”=Affiliation)
df.Author<-tibble::rowid_to_column(data.frame(df.Author), “AuthorId”) colnames(df.Author)

dbWriteTable(con,“Author”,df.Author, append=TRUE)

df.Author$ArticleId<-ArticleId

dbWriteTable(con,“Authorship”,df.Author[,c(7,1)], append=TRUE)

PubStatus

PubStatus<-c() PubMedDate<-c()

ExtractDate<-function(x) {

as.Date(paste(x[‘Year’],x[‘Month’], x[‘Day’], sep=“/”), format=“%Y/%m/%d”)

}

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“PubmedData”]][[“History”]])))) { PubStatus<-c(PubStatus,xmlAttrs(root_[[i]][[“PubmedData”]][[“History”]][[j]])) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][[“PubmedData”]][[“History”]])))) { PubMedDate<-c(PubMedDate,ExtractDate(xmlApply(root_[[i]][[“PubmedData”]][[“History”]][[j]],function(x)xmlValue(x))))

```
}
```

```
}

PubMedDate<-as.character(as.Date(PubMedDate, origin ="1970-01-01"))

ArticleId<-c()

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][["PubmedData"]][["History"]])))) {
ArticleId<-c(ArticleId,i)

 }

}

df.PubMed<-data.frame("ArticleId"=ArticleId,"PubStatus"=PubStatus,"PubMedDate"=PubMedDate) df.PubMed <-
tibble::rowid_to_column(data.frame(df.PubMed), "PubMedId")

dbWriteTable(con,"PubMedHistory",df.PubMed[,c(1,3,4)], append=TRUE)

dbWriteTable(con,"PubMed",df.PubMed[,c(2,1)], append=TRUE)
```

Grant

```
GrantId<-c() Acronym<-c() Agency<-c() Country<-c()

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][["MedlineCitation"]][["Article"]]
[["GrantList"]])))) { GrantId<-c(GrantId,xmlValue(root_[[i]][["MedlineCitation"]][["Article"]][["GrantList"]][[j]][["GrantID"]])
) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][["MedlineCitation"]][["Article"]]
[["GrantList"]])))) { Acronym<-c(Acronym,xmlValue(root_[[i]][["MedlineCitation"]][["Article"]][["GrantList"]][[j]]
[["Acronym"]]) ) } }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][["MedlineCitation"]][["Article"]]
[["GrantList"]])))) { Agency<-c(Agency,xmlValue(root_[[i]][["MedlineCitation"]][["Article"]][["GrantList"]][[j]][["Agency"]]) )
} }

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][["MedlineCitation"]][["Article"]]
[["GrantList"]])))) { Country<-c(Country,xmlValue(root_[[i]][["MedlineCitation"]][["Article"]][["GrantList"]][[j]][["Country"]])
) } }

ArticleId<-c()

for (i in seq(1:length(names(root_)))) { for (j in seq(1:length(names(root_[[i]][["MedlineCitation"]][["Article"]]
[["GrantList"]])))) { ArticleId<-c(ArticleId,i) } }

df.Grant<-data.frame("GrantId"=GrantId,"Acronym"=Acronym,"Agency"=Agency,"Country"=Country)

df.Grant$ArticleId<-ArticleId

library(dplyr) library(tidyr)

df.Grant<- df.Grant %>% drop_na()

df.Grant<-df.Grant[!duplicated(df.Grant[,c(1,5)]),]

df.Grant <-tibble::rowid_to_column(data.frame(df.Grant), "GrantIdNo")

dbWriteTable(con,"Grant",df.Grant[,-c(6)], append=TRUE)
```

dbWriteTable(con,“Grant_Ownership”,df.Grant[,c(6,1)], append=TRUE)

```
dbGetQuery(con, "select * from Author limit 5")
```

AuthorId <int>	LastName <chr>	ForeName <chr>	Initials <chr>	ValidYN <chr>
1	Kuo	Cassie	C	Y
2	Edwards	Alison	A	Y
3	Mazumdar	Madhu	M	Y
4	Memtsoudis	Stavros G	SG	Y
5	Stundner	Ottokar	O	Y

5 rows | 1-5 of 6 columns

```
dbGetQuery(con, "select * from Journal limit 5")
```

Issue_Id <int>	ISSN <chr>	IssnType <chr>	CitedMedium <chr>	Volume <chr>	Issue <chr>	PubDate <chr>
1	1556-3316	Print	Print	8	2	2012Jul
2	1545-7206	Electronic	Internet	54	2	2013 Mar-Apr
3	1524-4628	Electronic	Internet	43	11	2012Nov
4	1532-8651	Electronic	Internet	37	6	2012 Nov-Dec
5	1532-2688	Electronic	Internet	22	1	2013Jan

5 rows | 1-7 of 9 columns

```
dbGetQuery(con, "select * from Article limit 5")
```

ArticleId <chr>
1
2
3
4
5

5 rows | 1-1 of 5 columns

```
dbGetQuery(con, "select * from PubMedHistory limit 5")
```

PubMedId <int>	PubStatus <chr>	PubMedDate <chr>
1	received	2012-01-15
2	accepted	2012-04-16
3	epublish	2012-06-20
4	entrez	2013-07-23
5	pubmed	2013-07-23

5 rows

```
dbGetQuery(con, "select * from Grant limit 5")
```

GrantIdNo <int>	GrantId <chr>	Acronym <chr>	Agency <chr>	Country <chr>
1	U18 HS016-75	HS	AHRQ HHS	United States
2	UL1 RR024996	RR	NCRR NIH HHS	United States
3	UL1-RR024996	RR	NCRR NIH HHS	United States
4	RFA-HS-05-14	HS	AHRQ HHS	United States
5	UL1 TR000457	TR	NCATS NIH HHS	United States

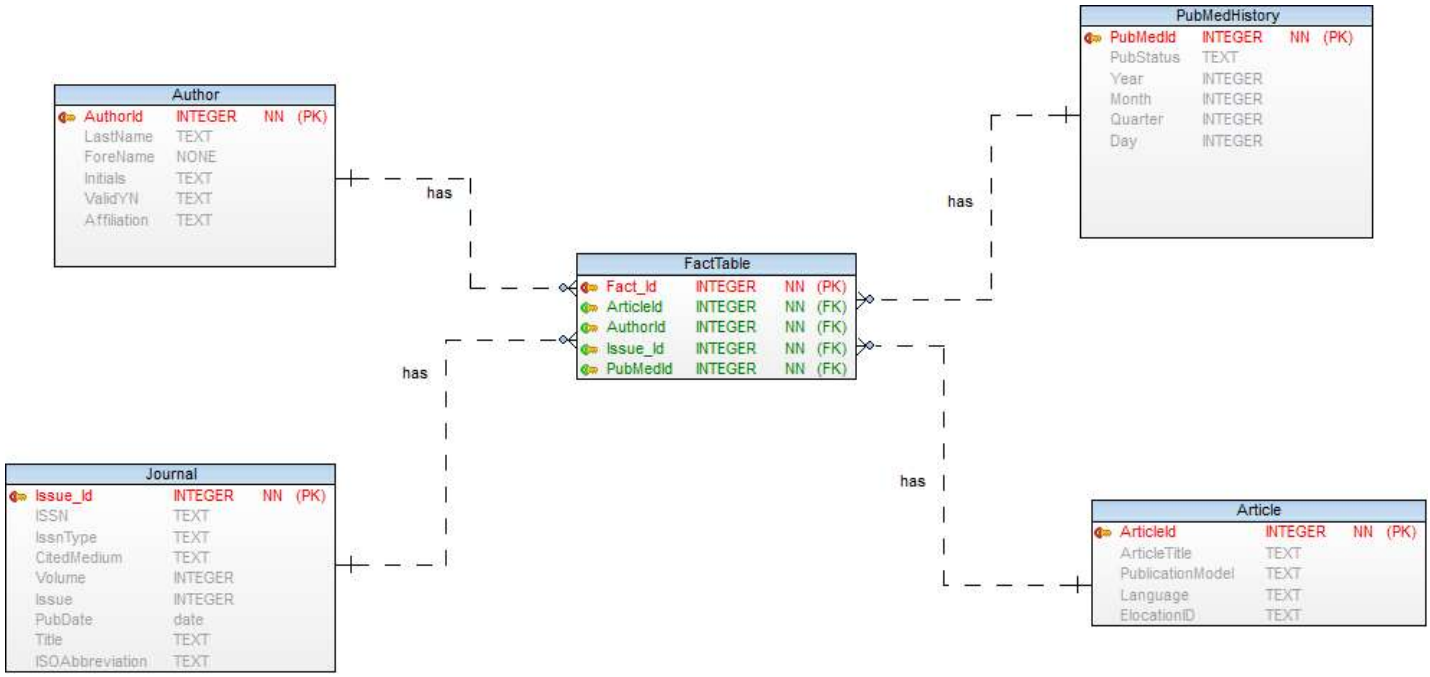
5 rows

```
dbGetQuery(con, "select * from PubMed limit 5")
```

ArticleId <chr>	PubMedId <int>
1	1
1	2
1	3
1	4
1	5

5 rows

Task 2



1. Create and populate a star schema with dimension and transaction fact tables. Each row in the fact table will represent one article. Include the image of an updated ERD that contains the fact table and any additional required dimension tables. Populate the star schema in R. When building the schema, look a head to Part 3 as the schema is dependent on the eventual OLAP queries.

```
dbDisconnect(con)
star <- dbConnect(RSQLite::SQLite(), "Star")
```

```
dbExecute(star, "PRAGMA foreign_keys = OFF;")
```

```
## [1] 0
```

dbExecute(star," CREATE TABLE Author (AuthorId INTEGER NOT NULL, LastName TEXT, ForeName NONE, Initials TEXT, ValidYN TEXT, Affiliation TEXT, CONSTRAINT PK_Author PRIMARY KEY (AuthorId)); ")

dbExecute(star," CREATE TABLE Journal (Issue_Id INTEGER NOT NULL, ISSN TEXT, IssnType TEXT, CitedMedium TEXT, Volume INTEGER, Issue INTEGER, PubDate date, Title TEXT, ISOAbbreviation TEXT, CONSTRAINT PK_Journal PRIMARY KEY (Issue_Id));

")

dbExecute(star," CREATE TABLE PubMedHistory (PubMedId INTEGER NOT NULL, PubStatus TEXT, Year INTEGER, Month INTEGER, Quarter INTEGER, Day INTEGER, CONSTRAINT PK_PubMedHistory PRIMARY KEY (PubMedId)); ")

dbExecute(star," CREATE TABLE Article (ArticleId INTEGER NOT NULL, ArticleTitle TEXT, PublicationModel TEXT, Language TEXT, ElocationID TEXT, CONSTRAINT PK_Article PRIMARY KEY (ArticleId)); ")

dbExecute(star," CREATE TABLE FactTable (Fact_Id INTEGER NOT NULL CONSTRAINT PK_FactTable PRIMARY KEY AUTOINCREMENT, ArticleId INTEGER NOT NULL, AuthorId INTEGER NOT NULL, Issue_Id INTEGER NOT NULL, PubMedId INTEGER NOT NULL, CONSTRAINT has FOREIGN KEY (AuthorId)

REFERENCES Author (AuthorId), CONSTRAINT has FOREIGN KEY (Issue_Id) REFERENCES Journal (Issue_Id), CONSTRAINT has FOREIGN KEY (PubMedId) REFERENCES PubMedHistory (PubMedId), CONSTRAINT has FOREIGN KEY (ArticleId) REFERENCES Article (ArticleId));

")

PubMedDim<-dbGetQuery(con,“select PubMedId,PubStatus,strftime(‘%Y’,PubMedDate) as ‘Year’, strftime(‘%m’,PubMedDate) as ‘Month’,CASE WHEN cast(strftime(‘%m’, PubMedDate) as integer) BETWEEN 1 AND 3 THEN 1 WHEN cast(strftime(‘%m’, PubMedDate) as integer) BETWEEN 4 and 6 THEN 2 WHEN cast(strftime(‘%m’, PubMedDate) as integer) BETWEEN 7 and 9 THEN 3 ELSE 4 END as Quarter,strftime(‘%d’,PubMedDate) as ‘Day’ from PubMedHistory”)

Fact<-dbGetQuery(con,“select Ar.ArticleId,A.AuthorId, J.Issue_Id,P.PubMedId from Article Ar inner join Authorship A on Ar.ArticleId=A.ArticleId inner join PubMed P on P.ArticleId=Ar.ArticleId inner join Journal_Ownership J on J.ArticleId=Ar.ArticleId”)

dbWriteTable(star,“Author”,df.Author[,-c(7)], append=TRUE)

dbWriteTable(star,“Journal”,df.Journal[,-c(10)], append=TRUE)

dbWriteTable(star,“Article”,df.Article, append=TRUE)

dbWriteTable(star,“PubMedHistory”,PubMedDim, append=TRUE)

dbWriteTable(star,“FactTable”,Fact, append=TRUE)

```
dbGetQuery(star,"select * from PubMedHistory limit 5")
```

PubMedId <int>	PubStatus <chr>	Year <int>	Month <int>	Quarter <int>	Day <int>
1	received	2012	1	1	15
2	accepted	2012	4	2	16
3	epublish	2012	6	2	20
4	entrez	2013	7	3	23
5	pubmed	2013	7	3	23

5 rows

```
dbGetQuery(star,"select * from Author limit 5")
```

AuthorId <int>	LastName <chr>	ForeName <chr>	Initials <chr>	ValidYN <chr>
1	Kuo	Cassie	C	Y
2	Edwards	Alison	A	Y
3	Mazumdar	Madhu	M	Y
4	Memtsoudis	Stavros G	SG	Y
5	Stundner	Ottokar	O	Y

5 rows | 1-5 of 6 columns

```
dbGetQuery(star,"select * from Journal limit 5")
```

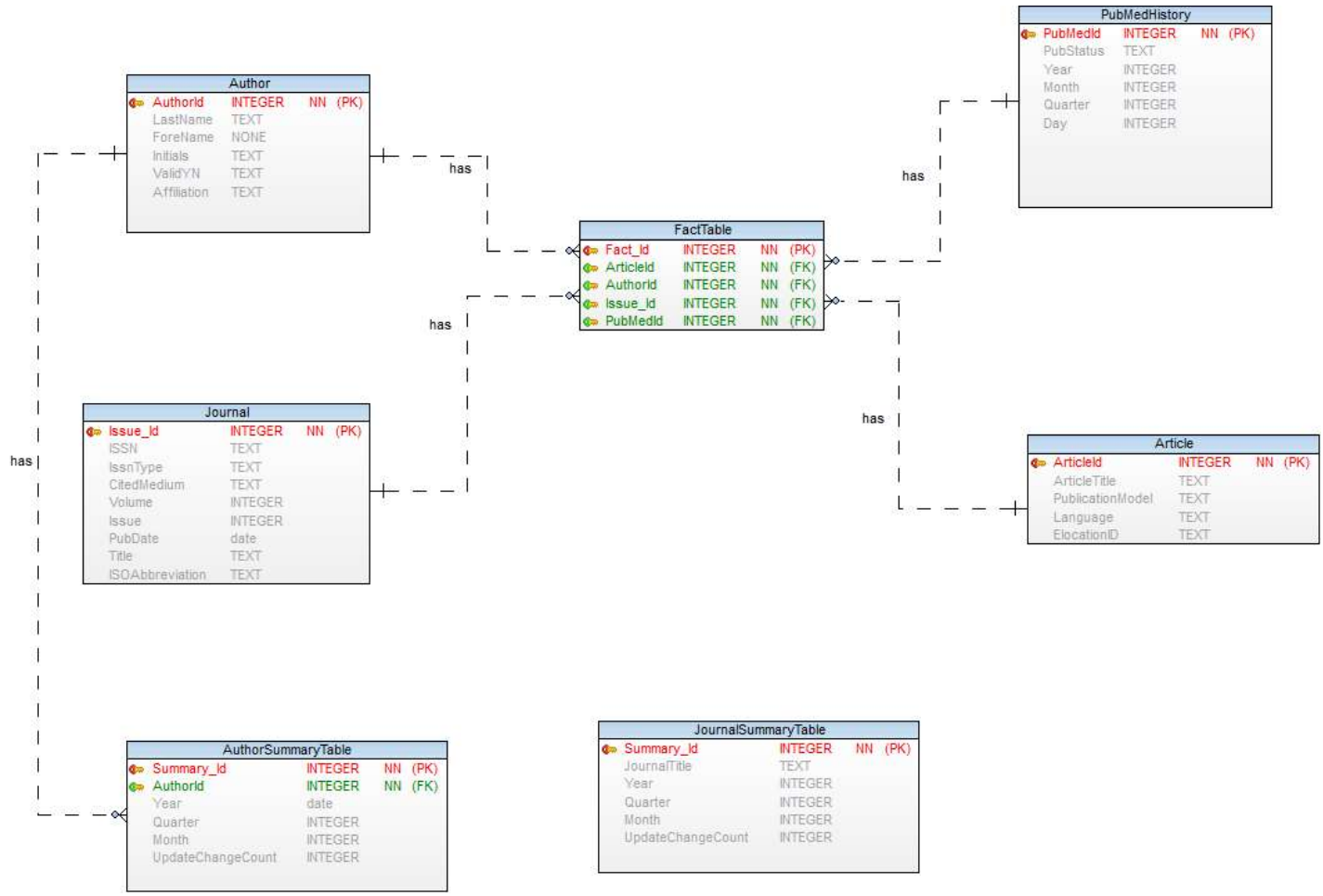
Issue_Id <int>	ISSN <chr>	IssnType <chr>	CitedMedium <chr>	Volume <int>	Issue <int>	PubDate <chr>
1	1556-3316	Print	Print	8	2	2012Jul
2	1545-7206	Electronic	Internet	54	2	2013 Mar-Apr
3	1524-4628	Electronic	Internet	43	11	2012Nov
4	1532-8651	Electronic	Internet	37	6	2012 Nov-Dec
5	1532-2688	Electronic	Internet	22	1	2013Jan

5 rows | 1-7 of 9 columns

```
select * from FactTable limit 5;
```

5 records

Fact_Id	ArticleId	AuthorId	Issue_Id	PubMedId
1	1	1	1	1
2	1	1	1	2
3	1	1	1	3
4	1	1	1	4
5	1	1	1	5



ERD Diagram Design

2. In the same schema as the previous step, create and populate a summary fact table that represents number of articles per time period (quarter, year) by author and by journal. Include the image of an updated ERD that contains the fact table. Populate the fact table in R. When building the schema, look a head to Part 3 as the schema is dependent on the eventual OLAP queries.

dbExecute(star,"

CREATE TABLE AuthorSummaryTable (Summary_Id INTEGER NOT NULL CONSTRAINT PK_AuthorSummaryTable PRIMARY KEY AUTOINCREMENT, AuthorId INTEGER NOT NULL, Year date, Quarter INTEGER, Month INTEGER, Day INTEGER, UpdateChangeCount INTEGER, CONSTRAINT has FOREIGN KEY (AuthorId) REFERENCES Author (AuthorId));

")

dbExecute(star,"

CREATE TABLE JournalSummaryTable (Summary_Id INTEGER NOT NULL CONSTRAINT PK_AuthorSummaryTable PRIMARY KEY AUTOINCREMENT, JournalTitle TEXT, Year INTEGER, Quarter INTEGER, Month INTEGER, Day INTEGER, UpdateChangeCount INTEGER

);

")

```
select  F.AuthorId,P.Year,P.Quarter,P.Month,P.Day ,count(F.ArticleId) as UpdateChangeCount
from FactTable as F, PubMedHistory as P where F.PubMedId=P.PubMedId Group by  F.AuthorId,
P.Year,P.Quarter,P.Month,P.Day
```

```
select J.Title as JournalTitle, P.Year,P.Quarter,P.Month,P.Day,count(F.ArticleId) as UpdateChange
Count
from Journal as J, PubMedHistory as P, FactTable as F where F.PubMedId=P.PubMedId
and F.Issue_Id=J.Issue_Id group by J.Title,P.Year,P.Quarter,P.Month,P.Day
```

dbWriteTable(star,“AuthorSummaryTable”,authorsummary, append=TRUE)

dbWriteTable(star,“JournalSummaryTable”,journalsummary, append=TRUE)

```
select * from AuthorSummaryTable limit 5;
```

5 records

Summary_Id	AuthorId	Year	Quarter	Month	Day	UpdateChangeCount
1	1	2012	1	1	15	1
2	1	2012	2	4	16	1
3	1	2012	2	6	20	1
4	1	2013	3	7	23	3
5	2	2012	1	1	15	1

```
select * from JournalSummaryTable limit 10;
```

Displaying records 1 - 10

Summary_Id	JournalTitle	Year	Quarter	Month	Day	UpdateChangeCount
1	Anesthesiology	2012	2	5	29	14
2	Anesthesiology	2013	3	7	9	7
3	BJU international	2011	3	8	18	11
4	BJU international	2011	3	8	20	22
5	BJU international	2012	2	4	24	11
6	Cancer	2011	2	6	27	10
7	Cancer	2011	3	8	26	10
8	Cancer	2011	3	9	19	10
9	Cancer	2011	4	10	21	10

Summary_Id	JournalTitle	Year	Quarter	Month	Day	UpdateChangeCount
10	Cancer	2011	4	10	25	20

Task 3

```
select JournalTitle,Year,Quarter,sum(UpdateChangeCount) from JournalSummaryTable group by Journal
Title,Year,Quarter order by Year,Quarter limit 20;
```

Displaying records 1 - 10

JournalTitle	Year	Quarter	sum(UpdateChangeCount)
Spine	2011	1	12
The Journal of arthroplasty	2011	1	6
Cancer	2011	2	10
Journal of clinical anesthesia	2011	2	6
BJU international	2011	3	33
Cancer	2011	3	20
Journal of clinical anesthesia	2011	3	6
Journal of intensive care medicine	2011	3	18
PloS one	2011	3	12
The Journal of arthroplasty	2011	3	5

Part 3

1. Write queries using your data warehouse to explore whether the publications show a seasonal pattern. Look beyond the pattern of number of publications per season. Adjust your fact tables as needed to support your new queries. If you need to update the fact table, document your changes and your reasons why the changes are needed.

```
select AST.AuthorId,A.ForeName||" "||A.LastName as NAME,AST.Year,AST.Quarter,AST.Month,AST.Day, A
ST.UpdateChangeCount
                        from AuthorSummaryTable AST inner join Author A on AST.AuthorId=A.Autho
rID
```

```
head(AuthorPattern)
```

	AuthorId <int>	NAME <chr>	Year <int>	Quarter <int>	Month <int>	D... <int>	UpdateChangeCount <int>
1	1	Cassie Kuo	2012	1	1	15	1
2	1	Cassie Kuo	2012	2	4	16	1

	AuthorId <int>	NAME <chr>	Year <int>	Quarter <int>	Month <int>	D… <int>	UpdateChangeCount <int>
3	1	Cassie Kuo	2012	2	6	20	1
4	1	Cassie Kuo	2013	3	7	23	3
5	2	Alison Edwards	2012	1	1	15	1
6	2	Alison Edwards	2012	2	4	16	1

6 rows

2. visualize (graph/plot) the data from the previous step using R to explore seasonality and explain what you found.

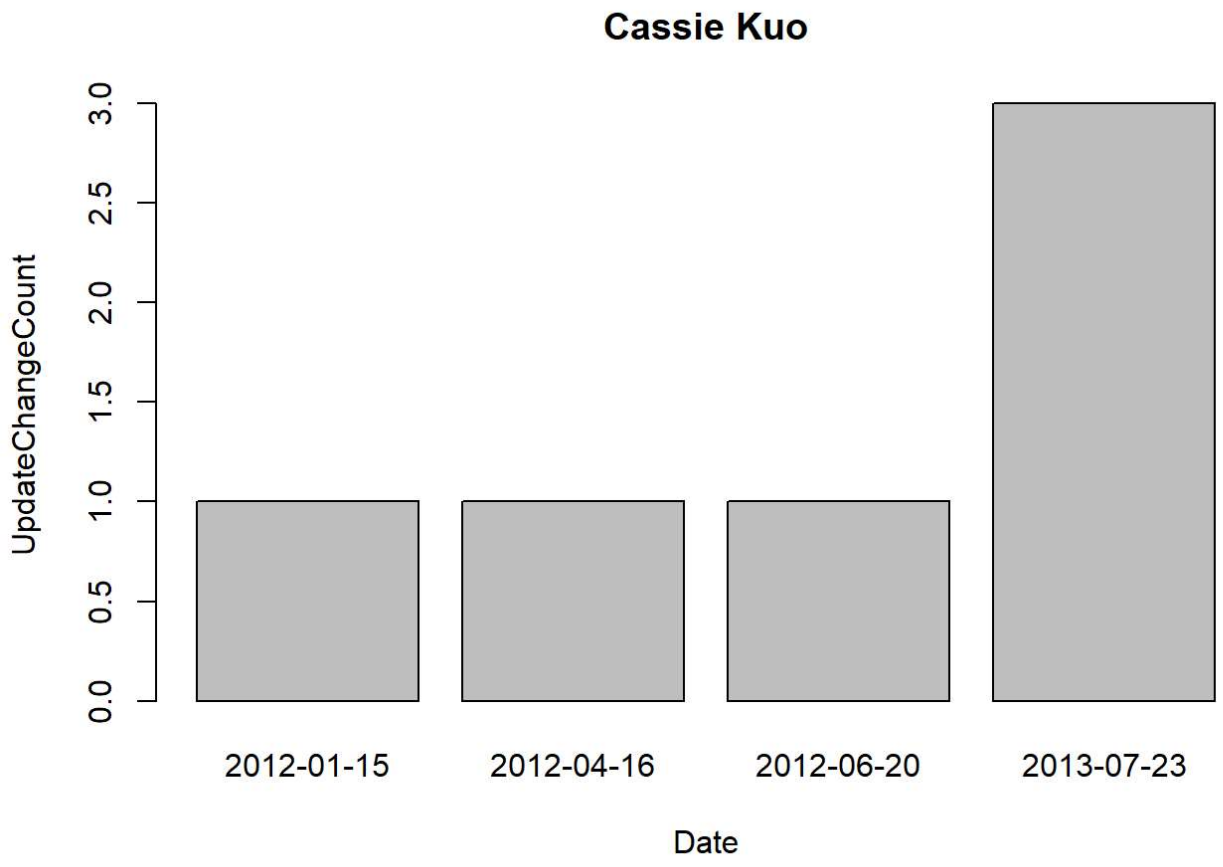
```
AuthorPattern$Date<-  as.Date(paste(AuthorPattern$Year,"-",AuthorPattern$Month,"-",AuthorPattern
$Day),format = '%Y - %m - %d')

head(AuthorPattern)
```

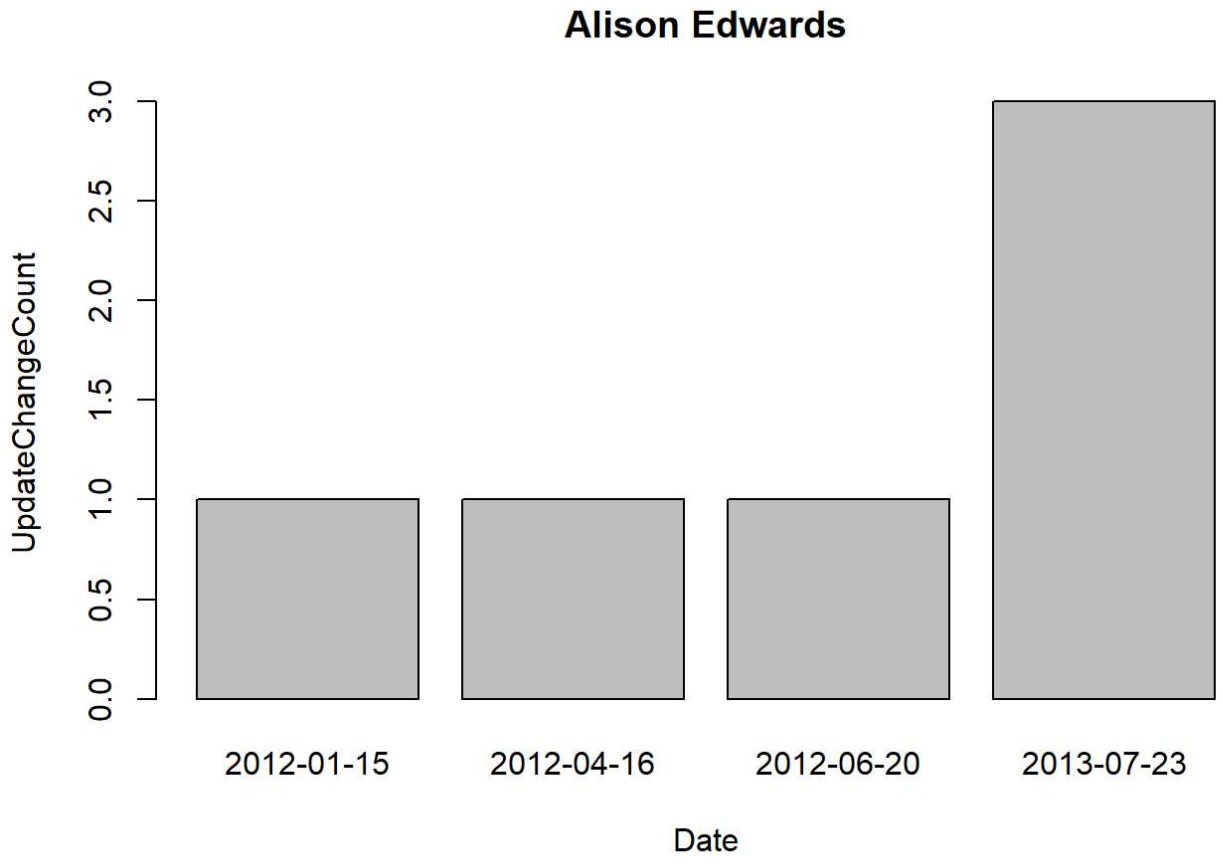
	AuthorId <int>	NAME <chr>	Year <int>	Quarter <int>	Mo… <int>	… <int>	UpdateChangeCount <int>	Date <date>
1	1	Cassie Kuo	2012	1	1	15	1	2012-01-15
2	1	Cassie Kuo	2012	2	4	16	1	2012-04-16
3	1	Cassie Kuo	2012	2	6	20	1	2012-06-20
4	1	Cassie Kuo	2013	3	7	23	3	2013-07-23
5	2	Alison Edwards	2012	1	1	15	1	2012-01-15
6	2	Alison Edwards	2012	2	4	16	1	2012-04-16

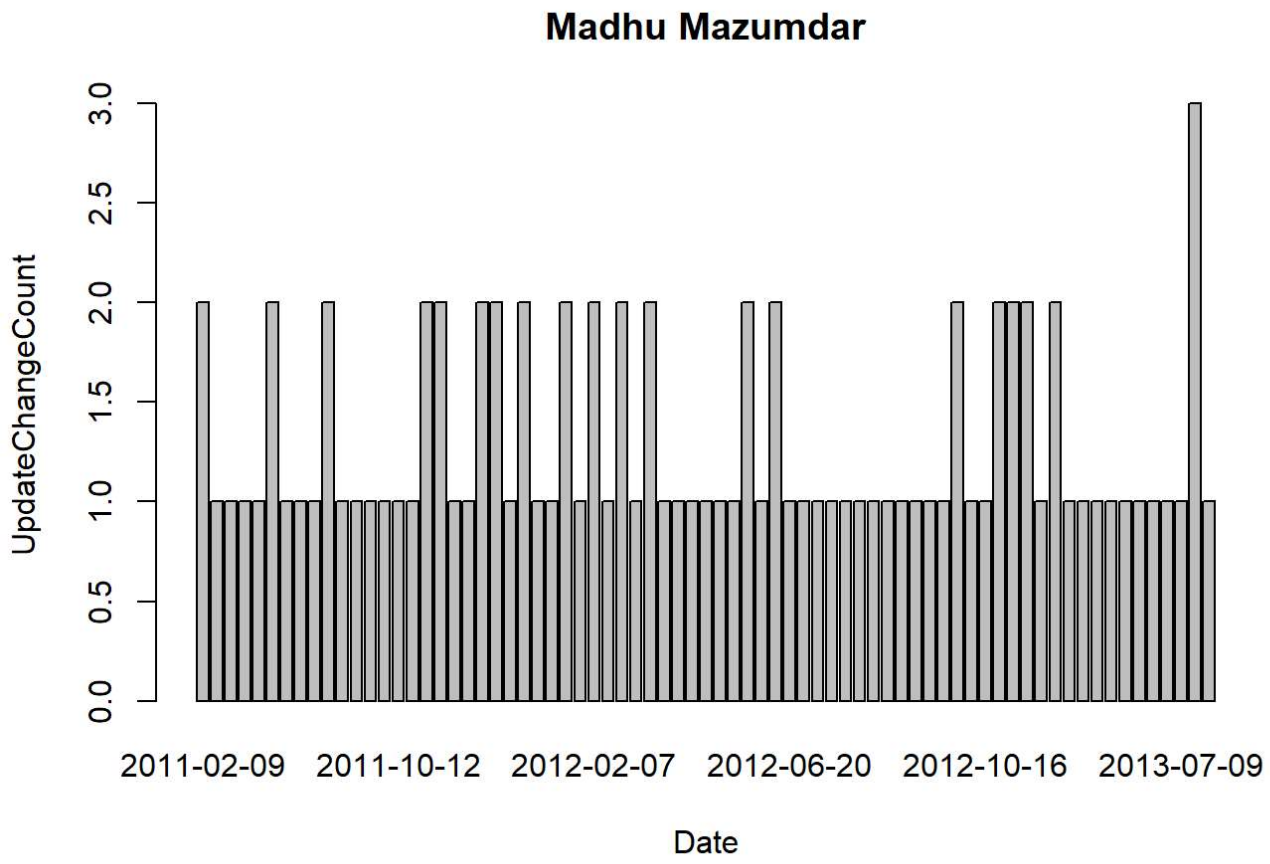
6 rows

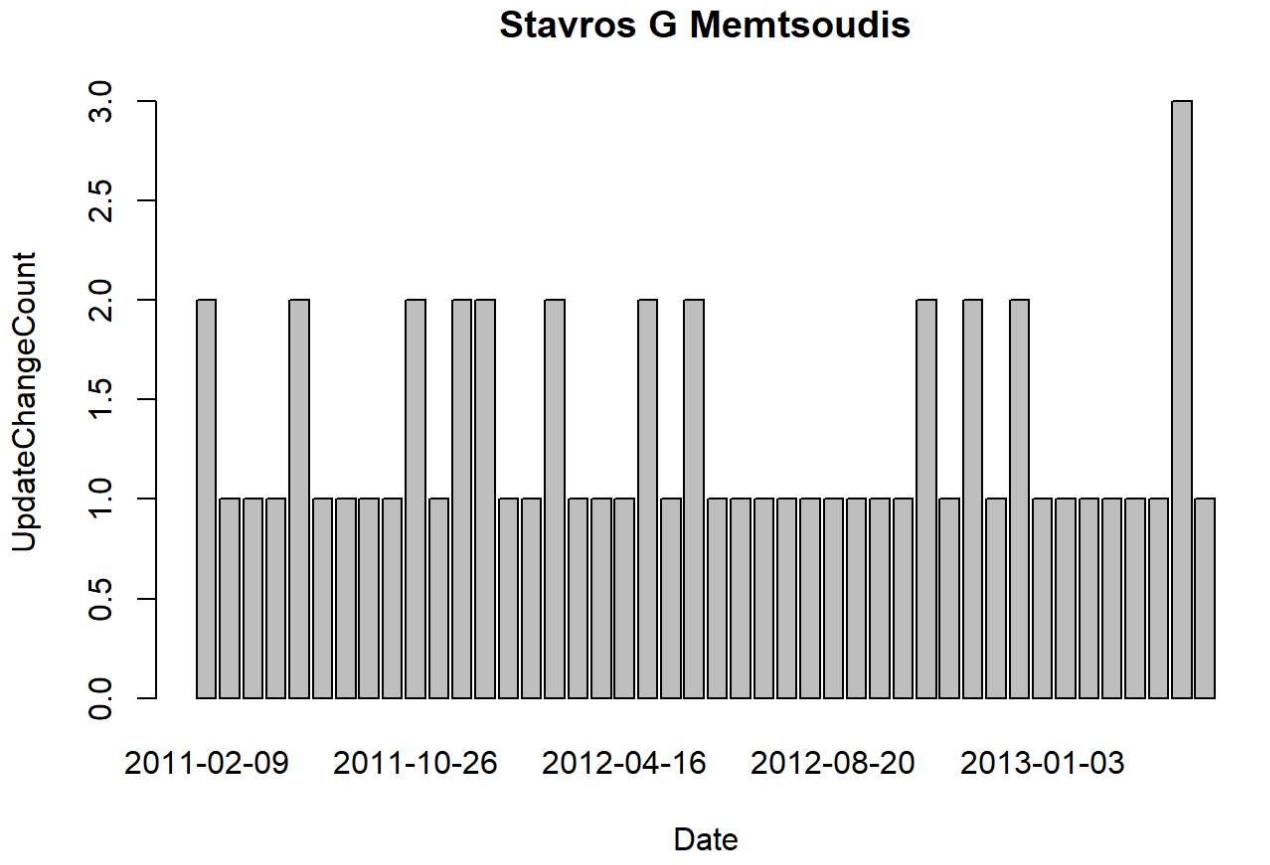
2. Either (a) visualize (graph/plot) the data from the previous step using R to explore seasonality and explain what you found, or (b) build a predictive model to forecast the expected number of publications for a quarter. (Note that we do not cover predictive modeling in this course, so if you do not know this from a prior course, then simply create the visualization.)

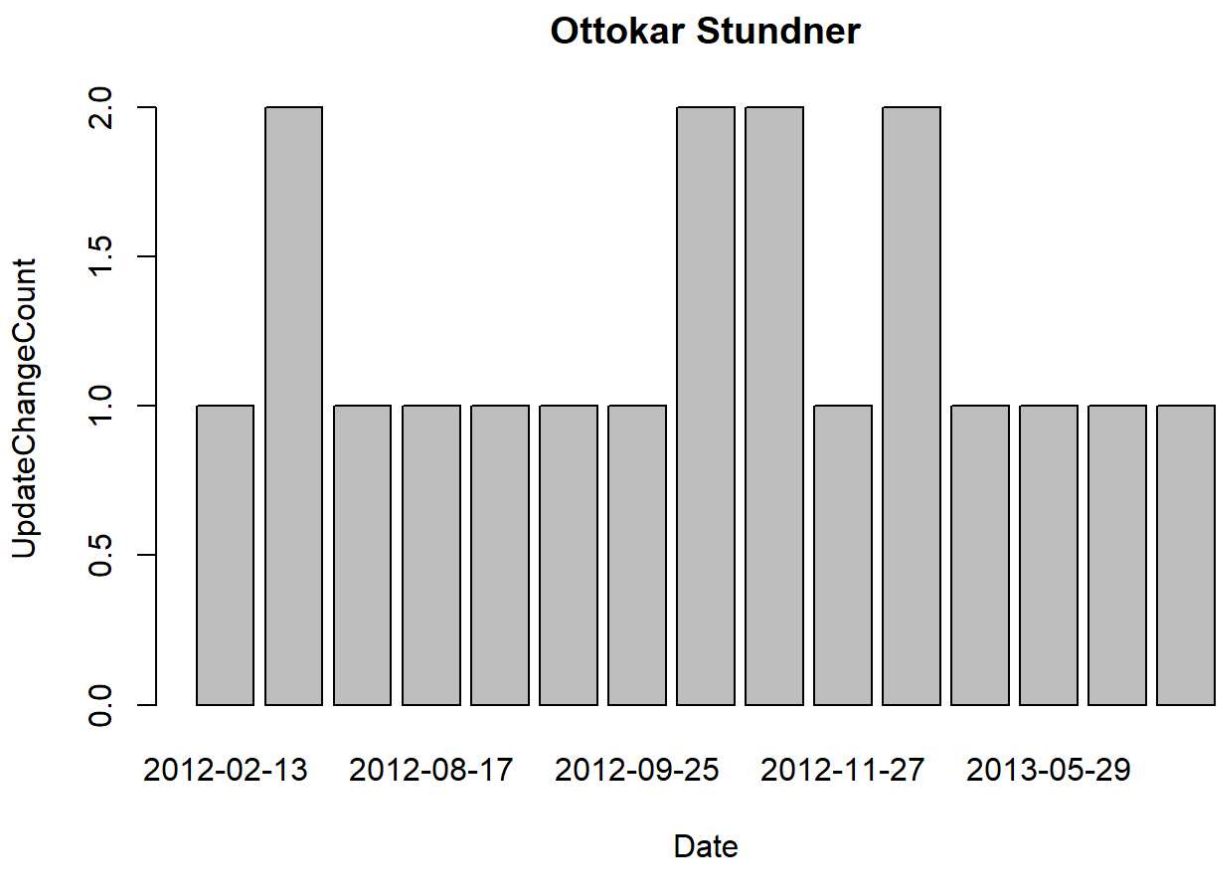
```
for (i in (unique(AuthorPattern$NAME))){
      barplot(UpdateChangeCount ~ Date, data = AuthorPattern[which(AuthorPattern$NAME==i),],main=
i)
      }
```

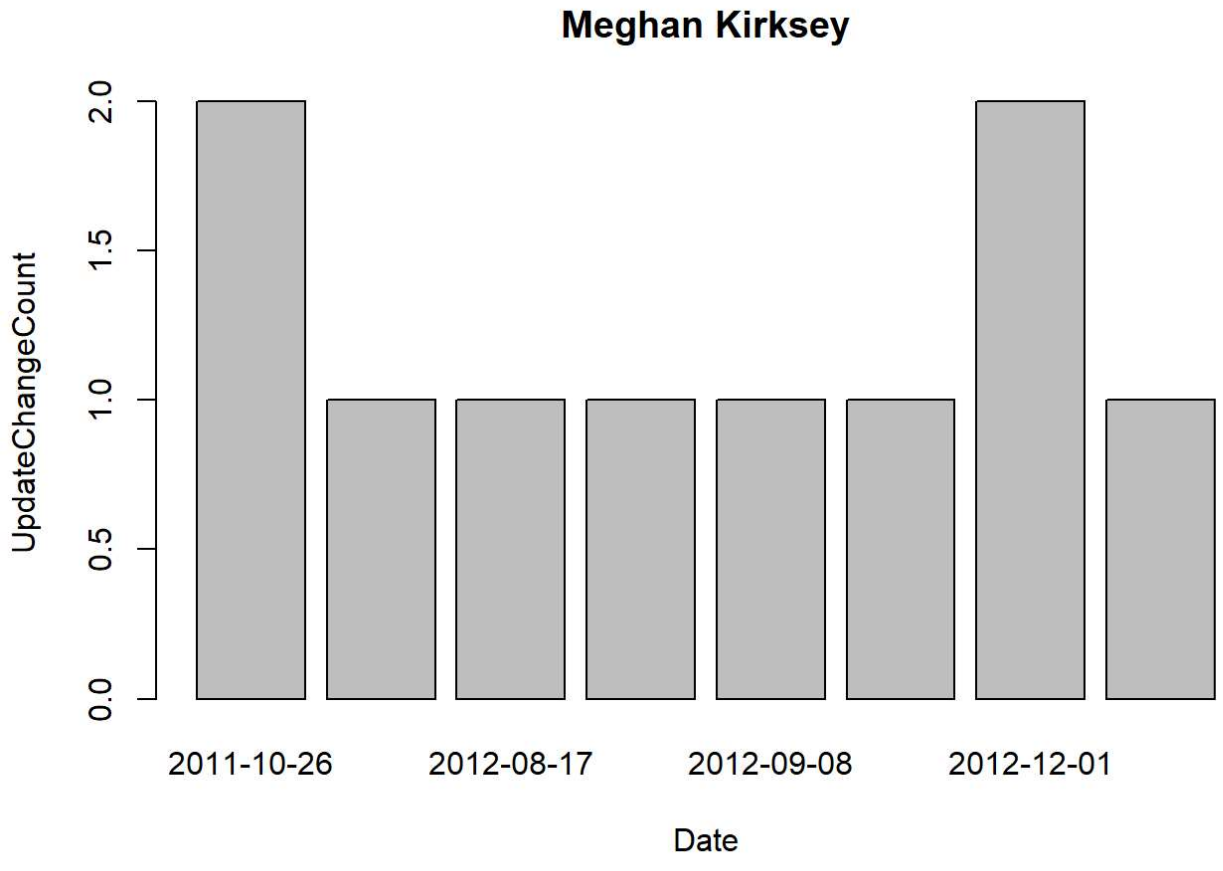

Cassie Kuo
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2012-01-15
2012-04-16
2012-06-20
2013-07-23
Date

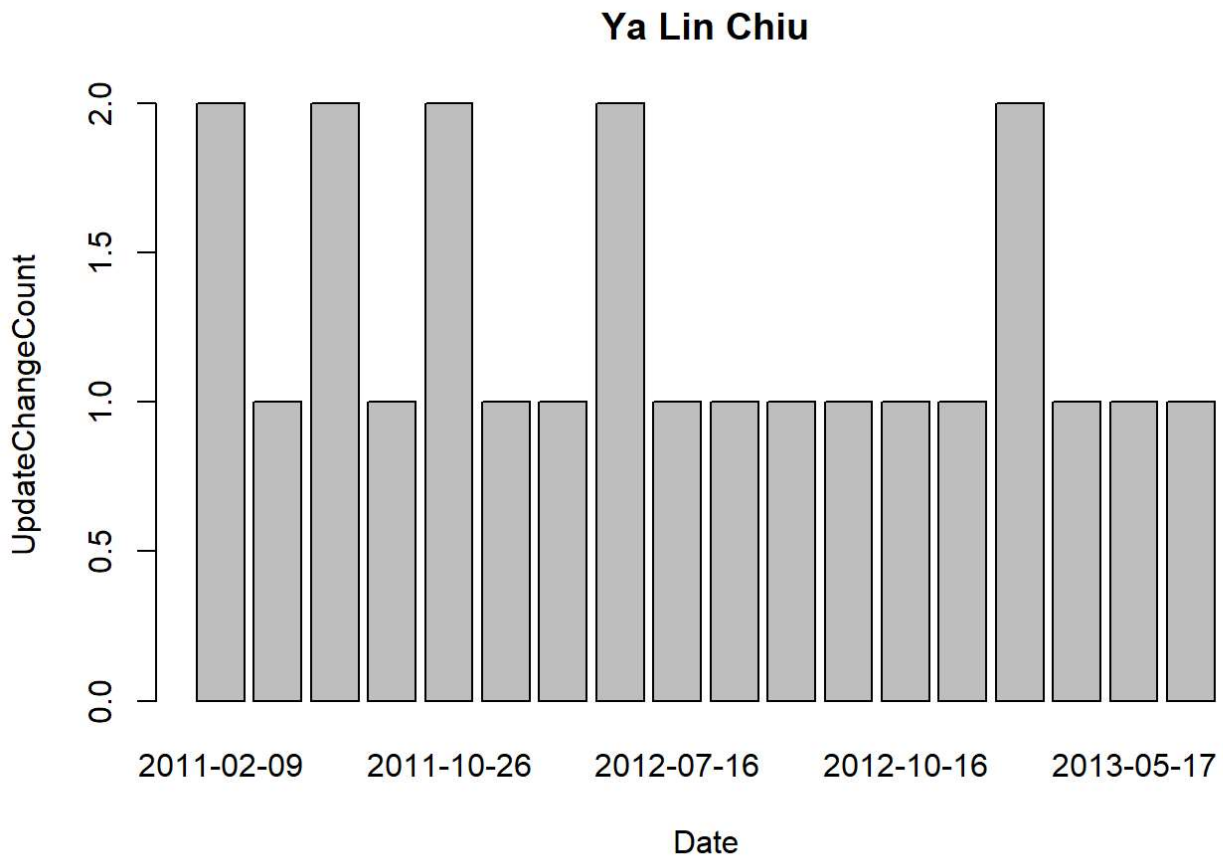

Alison Edwards
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2012-01-15
2012-04-16
2012-06-20
2013-07-23
Date

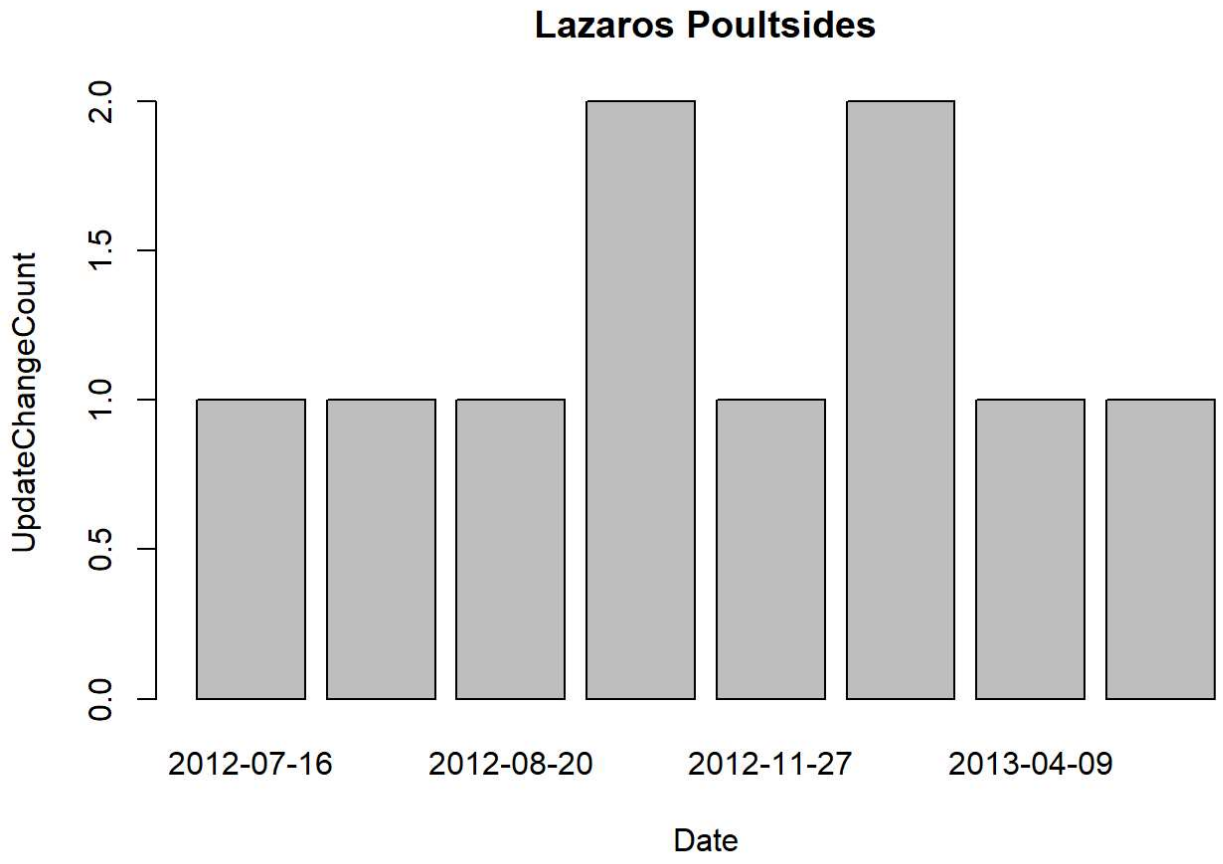

Madhu Mazumdar
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2011-02-09
2011-10-12
2012-02-07
2012-06-20
2012-10-16
2013-07-09
Date

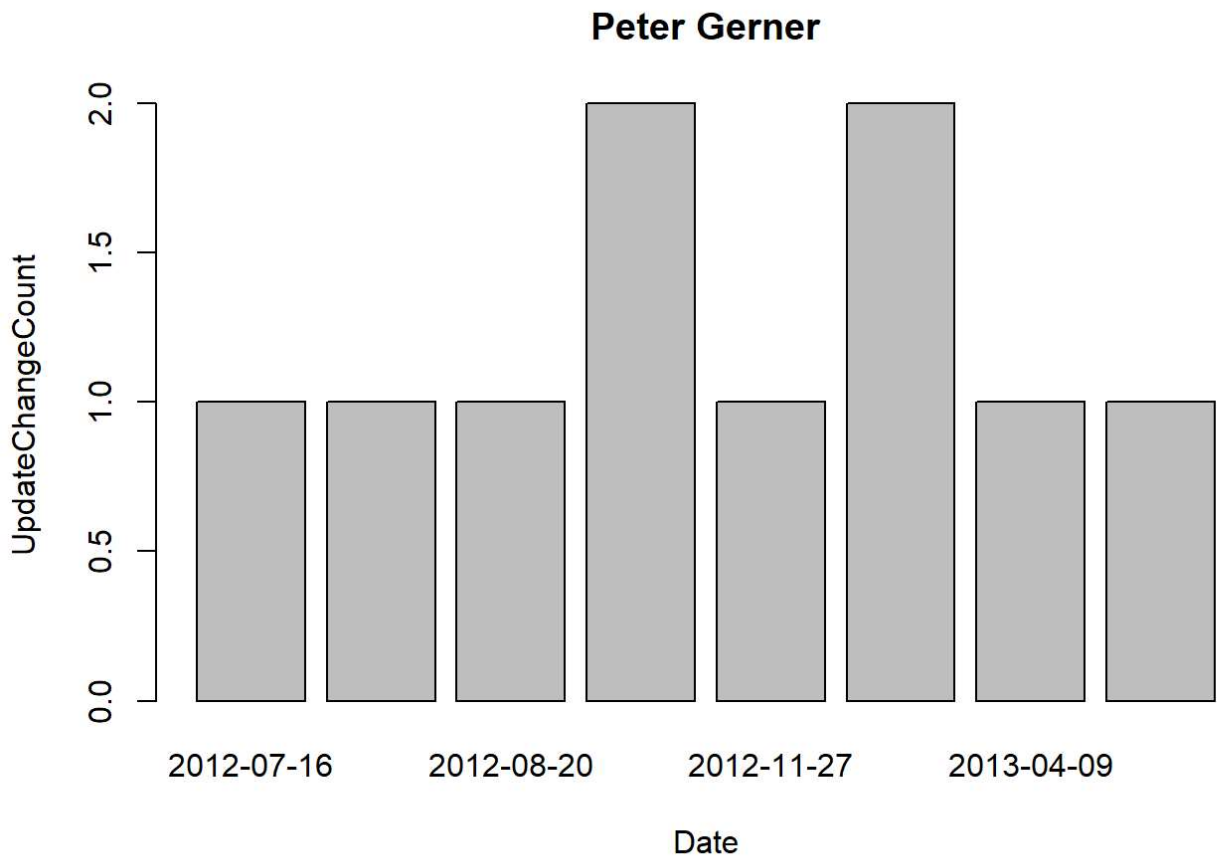

Stavros G Memtsoudis
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2.5
3.0
2011-02-09
2011-10-26
2012-04-16
2012-08-20
2013-01-03
Date

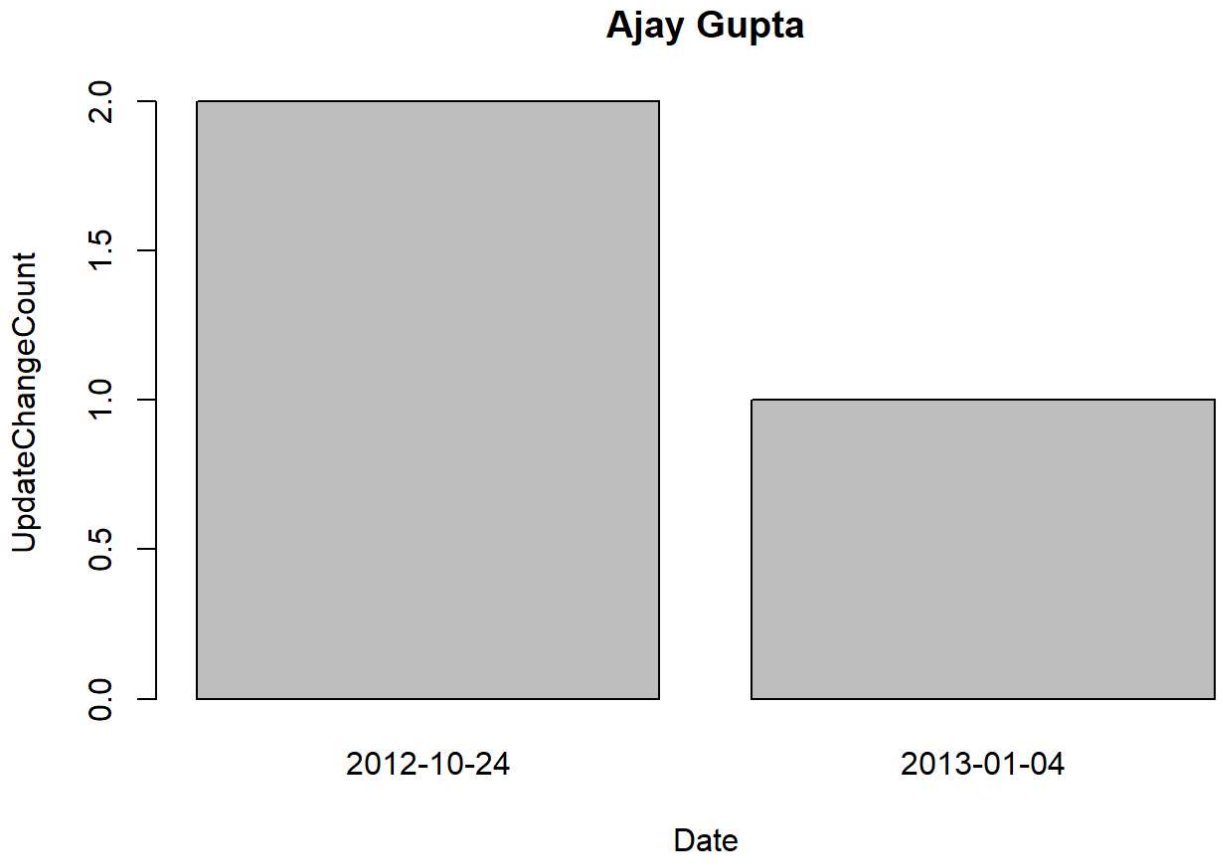

Ottokar Stundner
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-02-13
2012-08-17
2012-09-25
2012-11-27
2013-05-29
Date

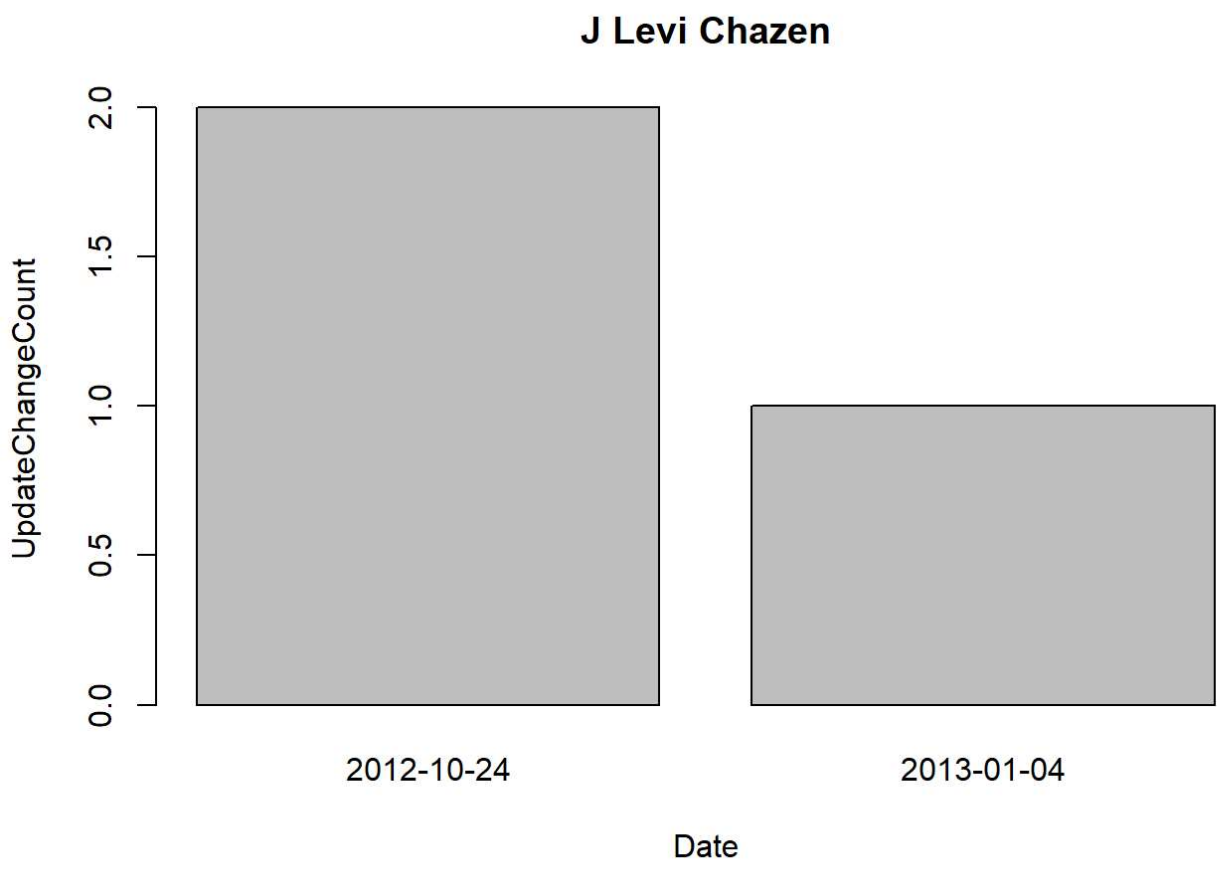

Meghan Kirksey
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-10-26
2012-08-17
2012-09-08
2012-12-01
Date

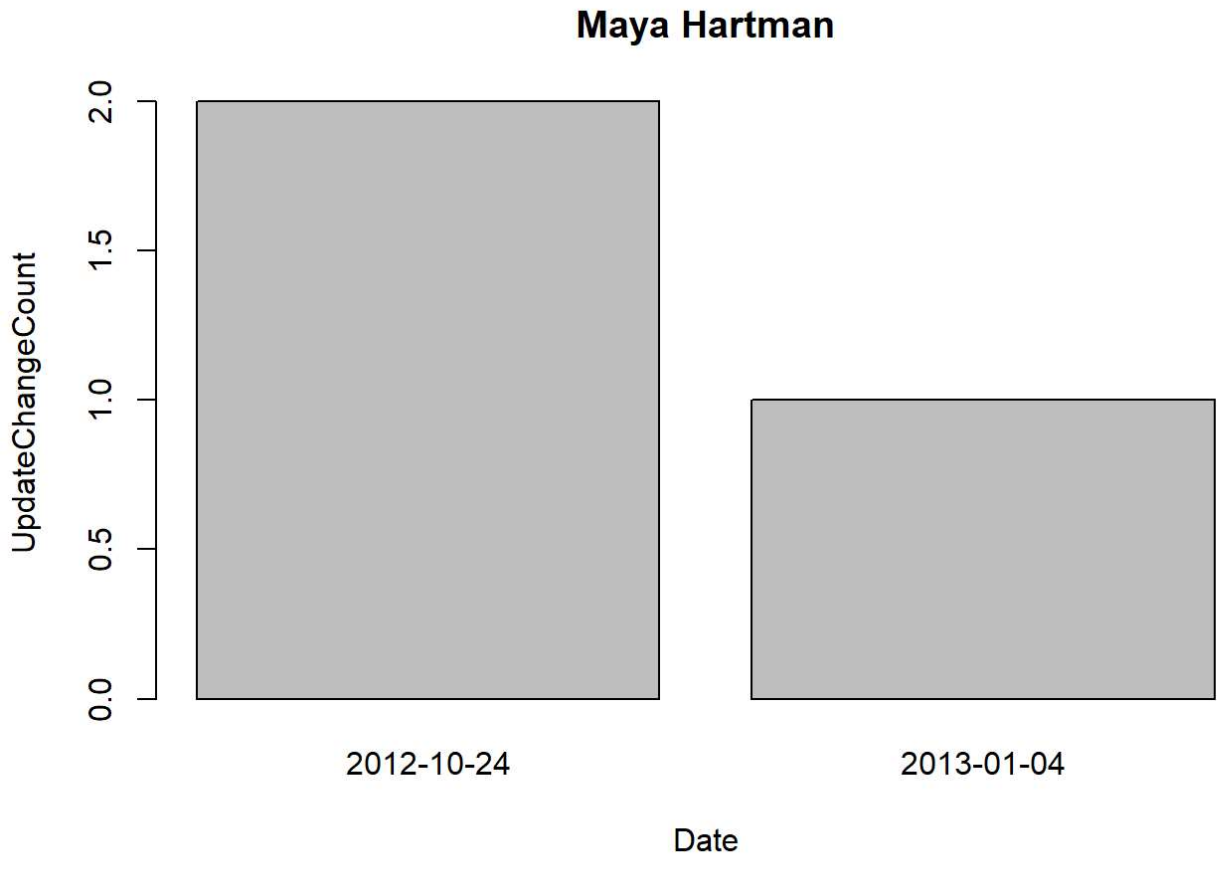

Ya Lin Chiu
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-02-09
2011-10-26
2012-07-16
2012-10-16
2013-05-17
Date

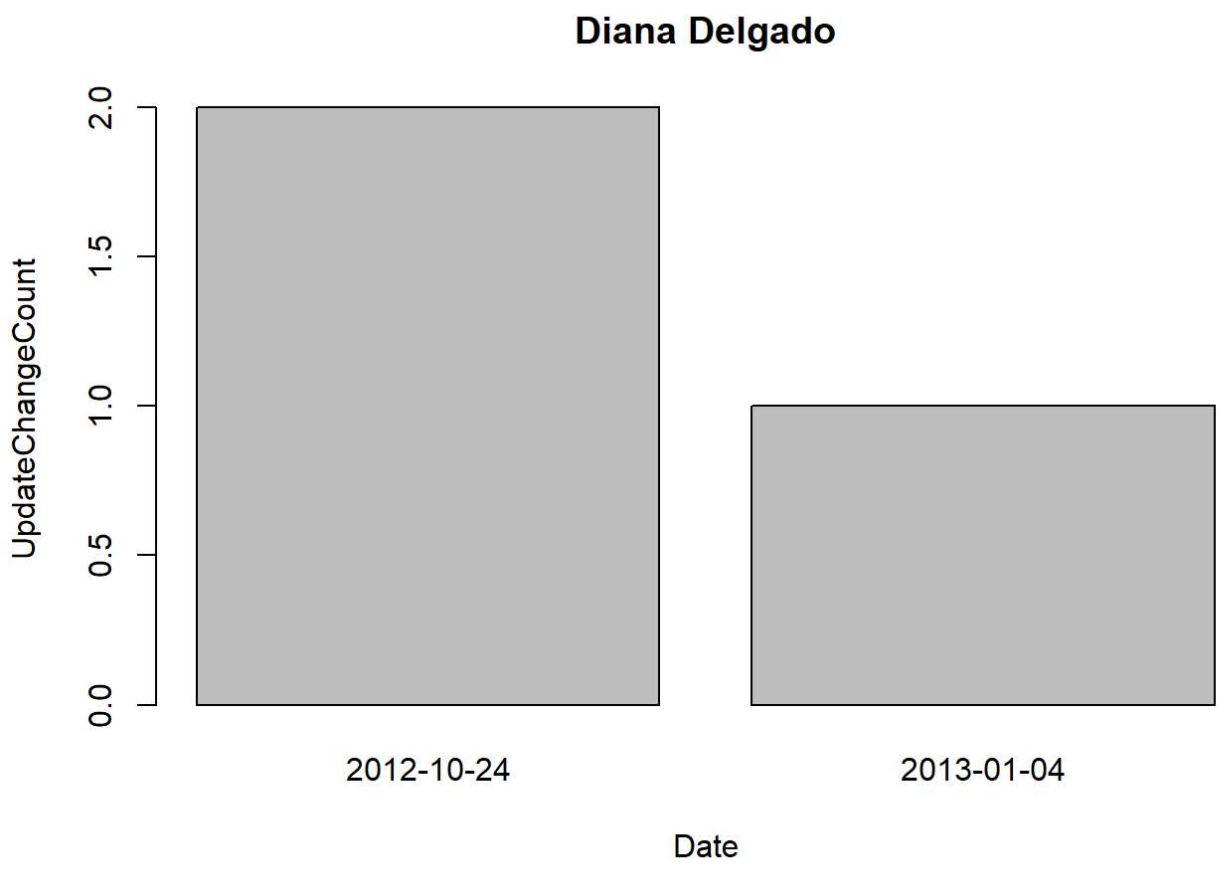

Lazaros Poultsides
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-07-16
2012-08-20
2012-11-27
2013-04-09
Date

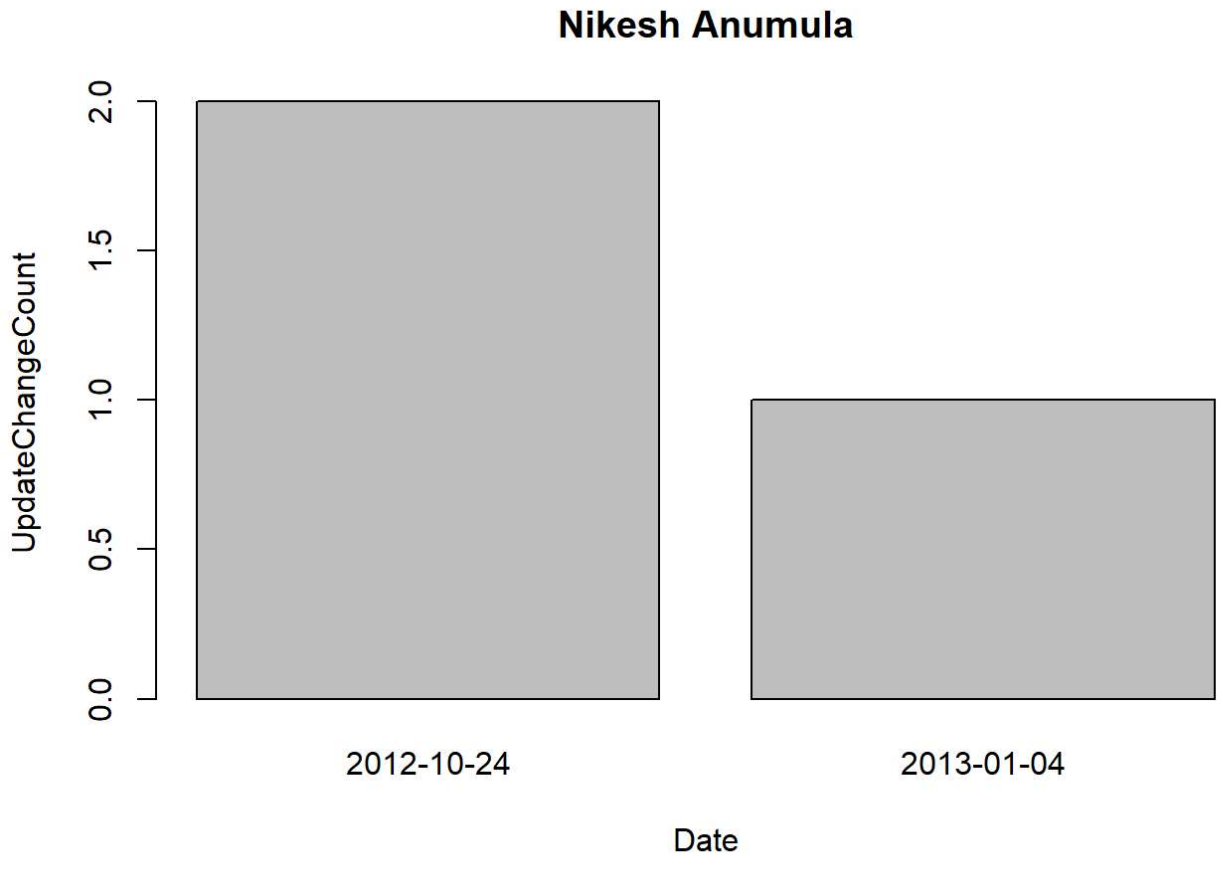

Peter Gerner
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-07-16
2012-08-20
2012-11-27
2013-04-09
Date

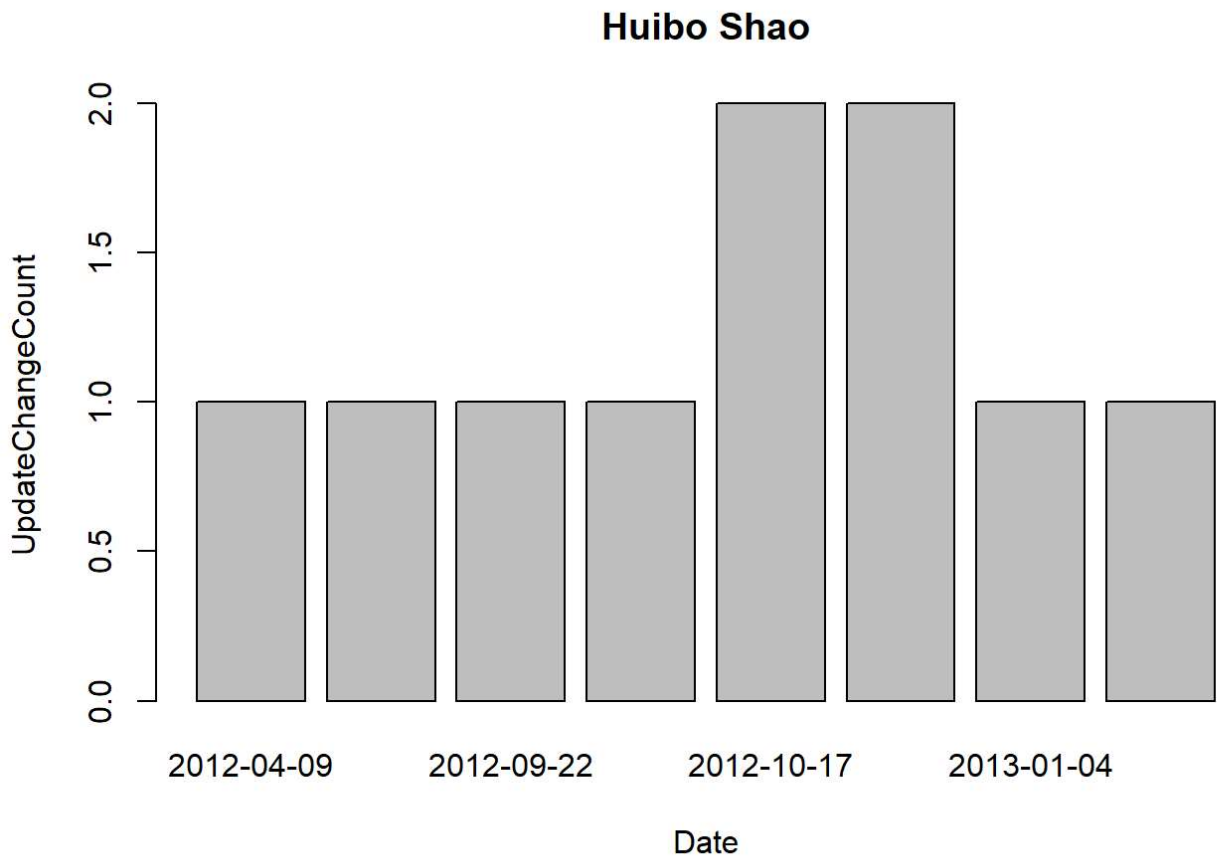

Ajay Gupta
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

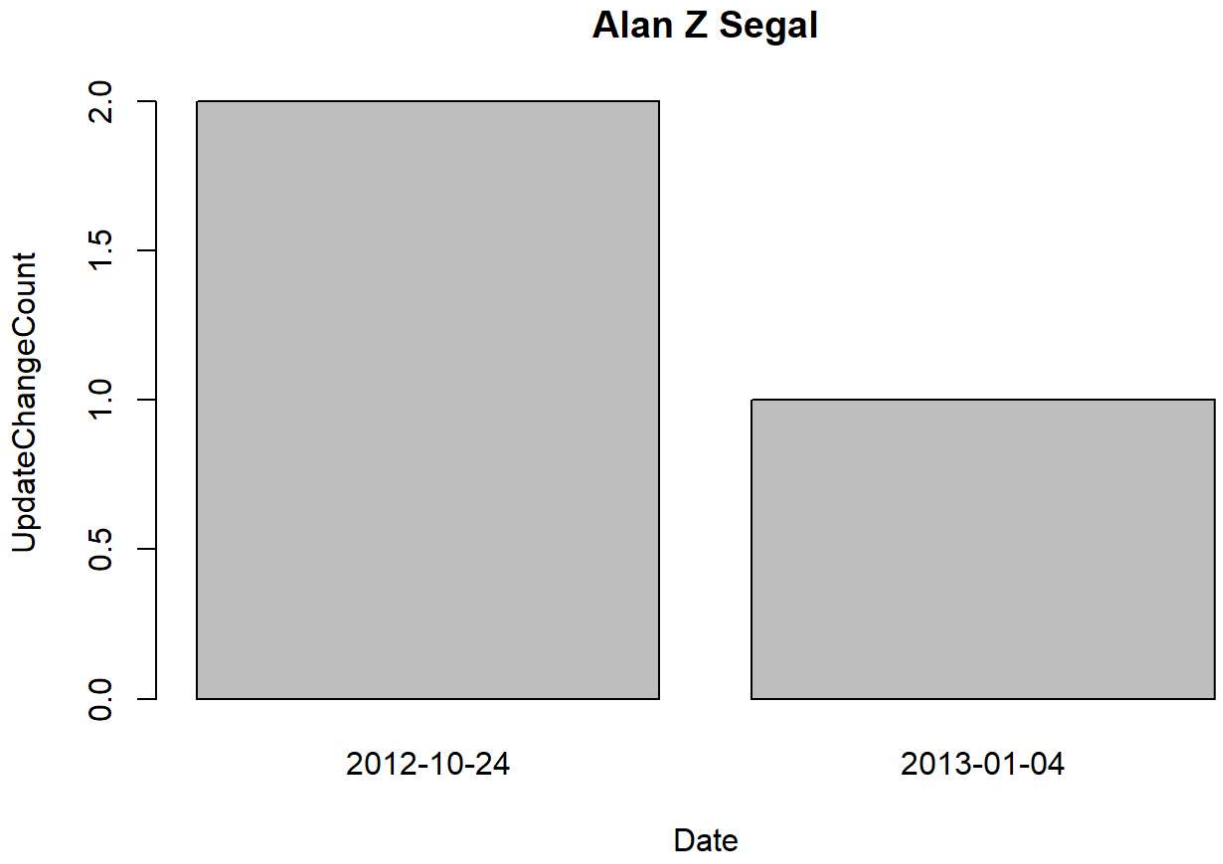

J Levi Chazen
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

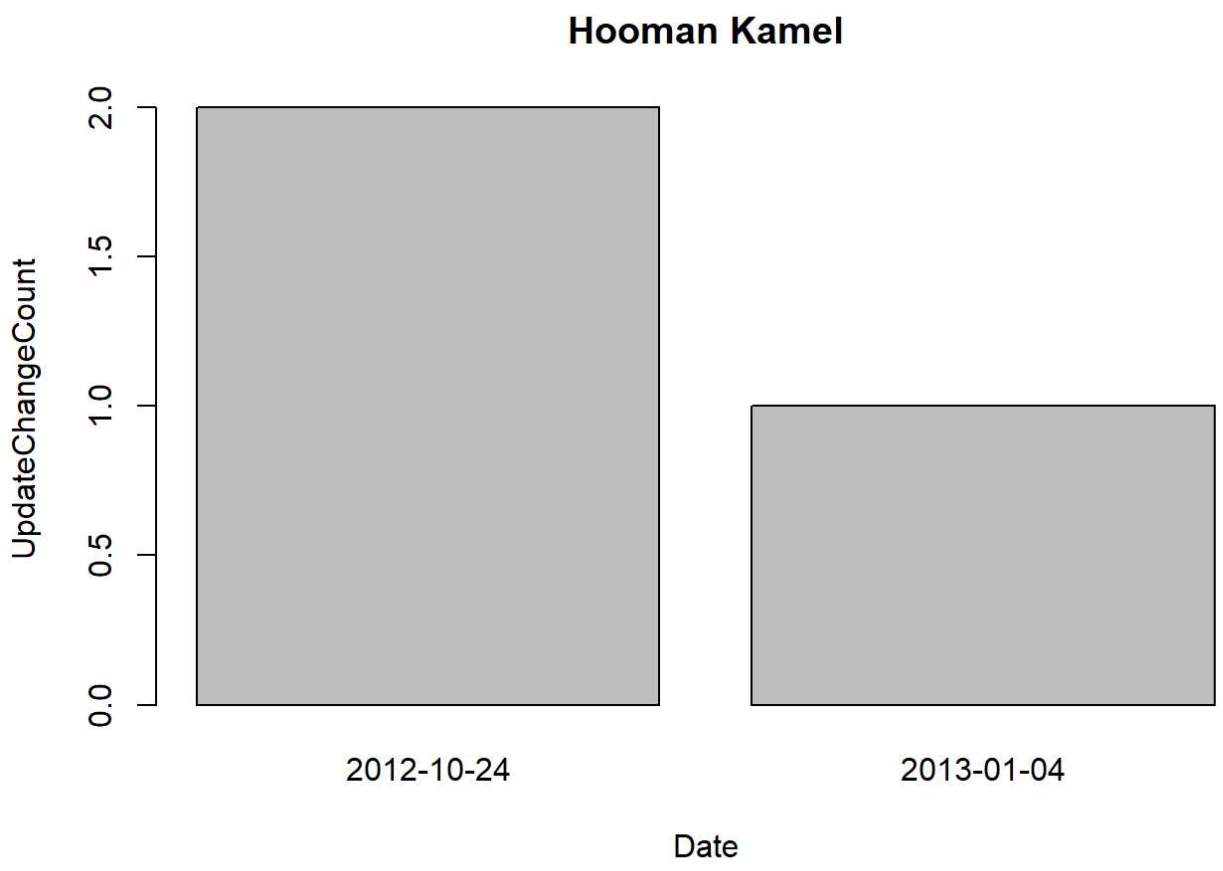

Maya Hartman
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

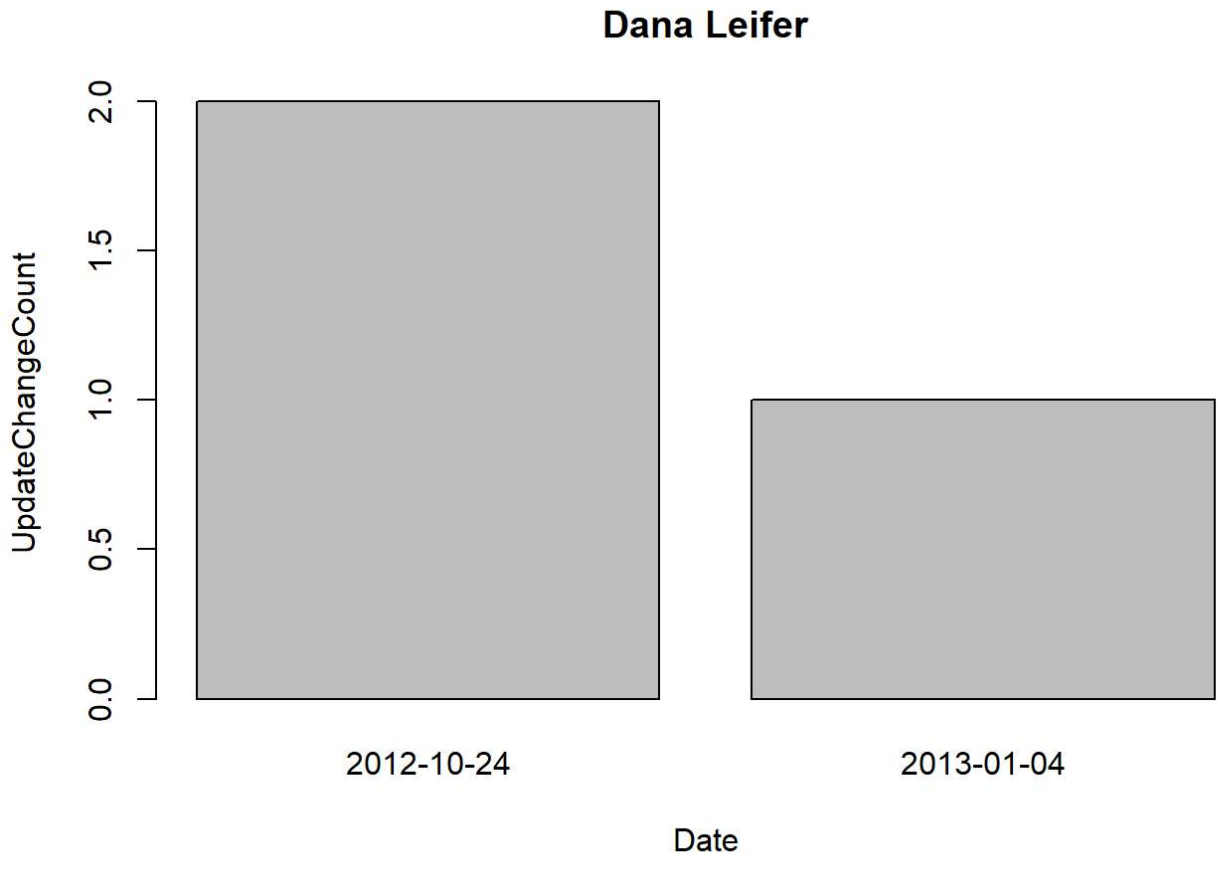

Diana Delgado
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

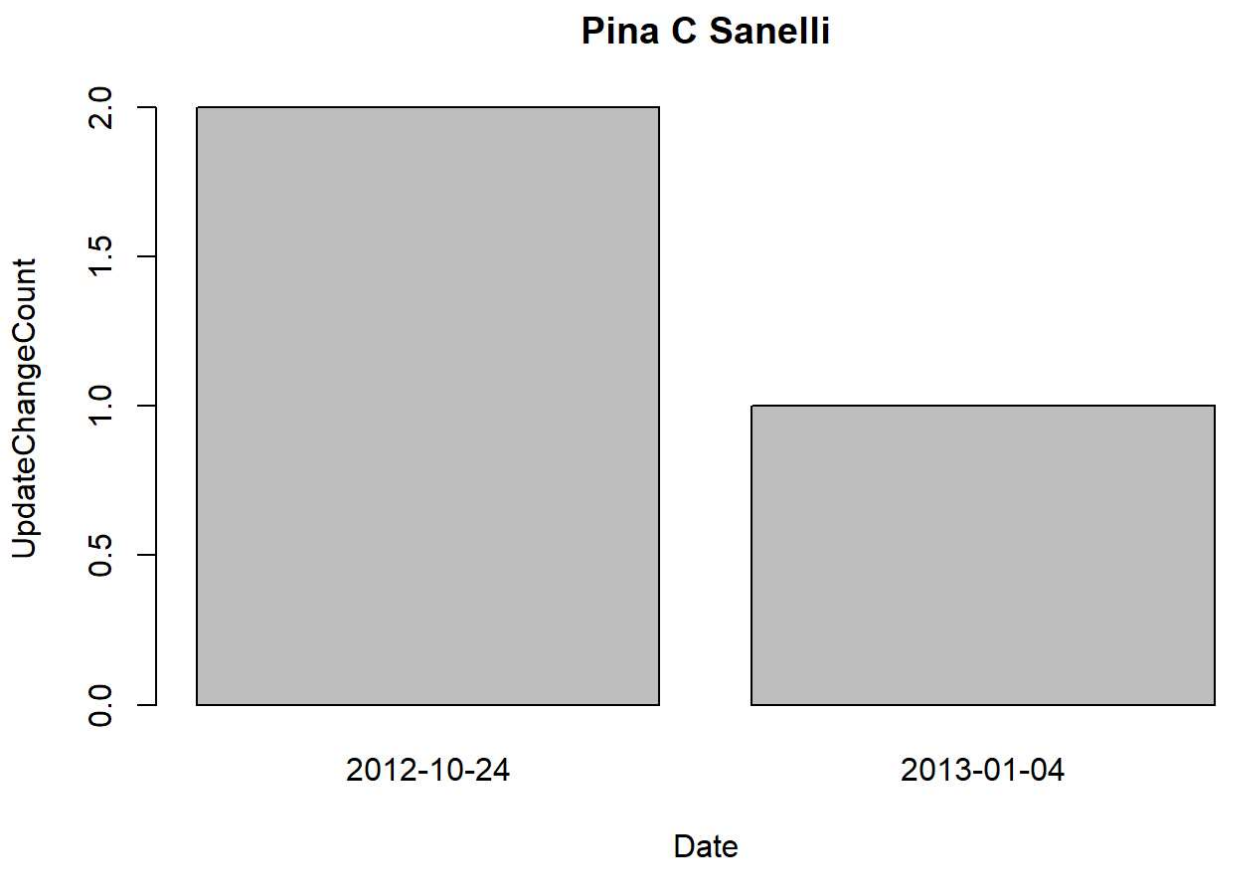

Nikesh Anumula
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

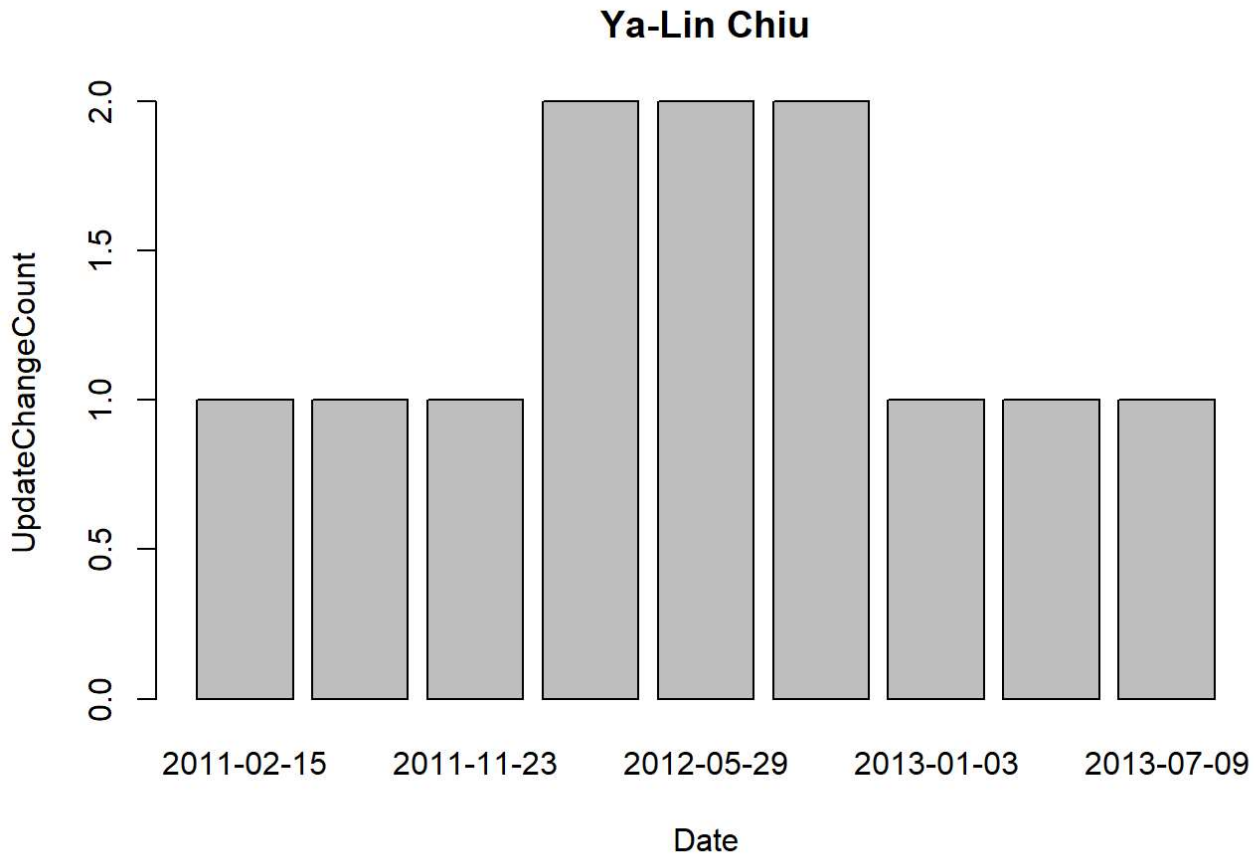

Huibo Shao
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-04-09
2012-09-22
2012-10-17
2013-01-04
Date

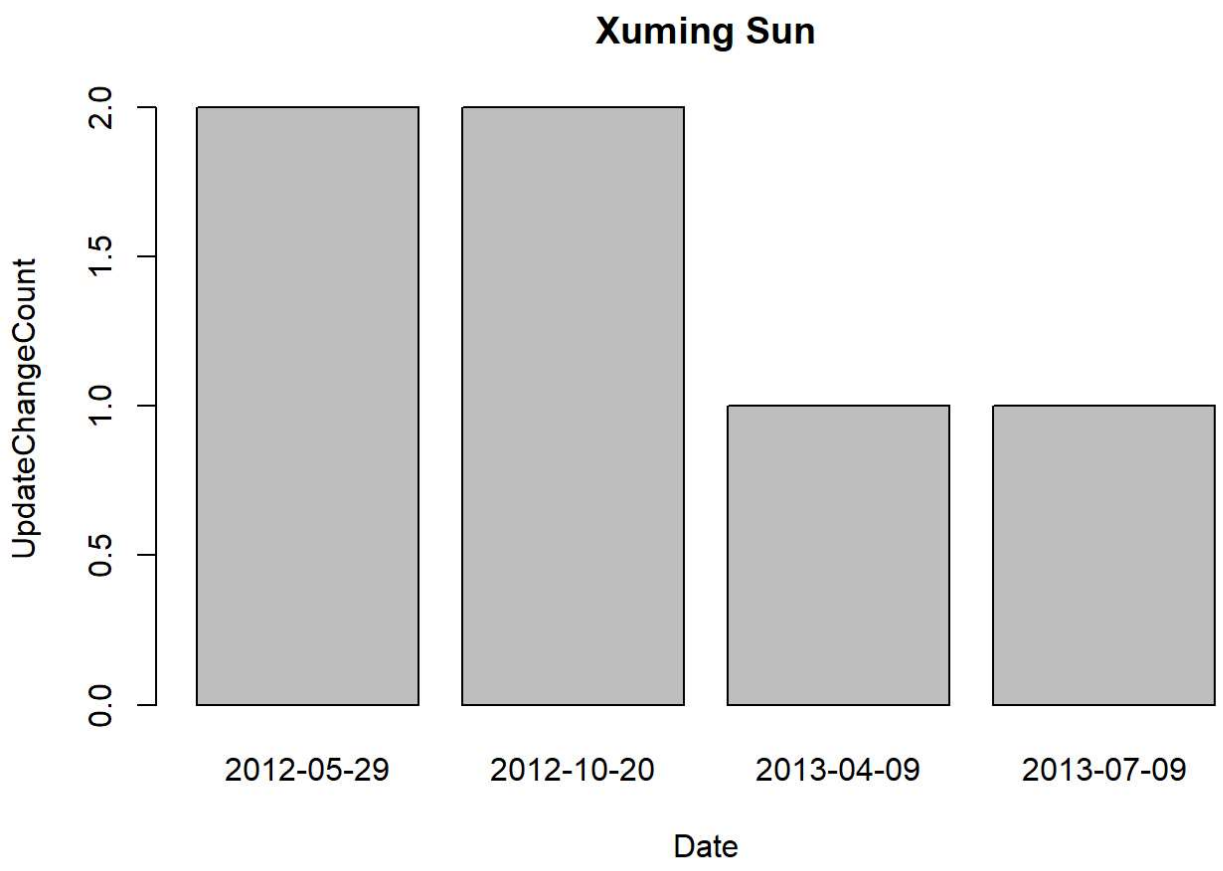

Alan Z Segal
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

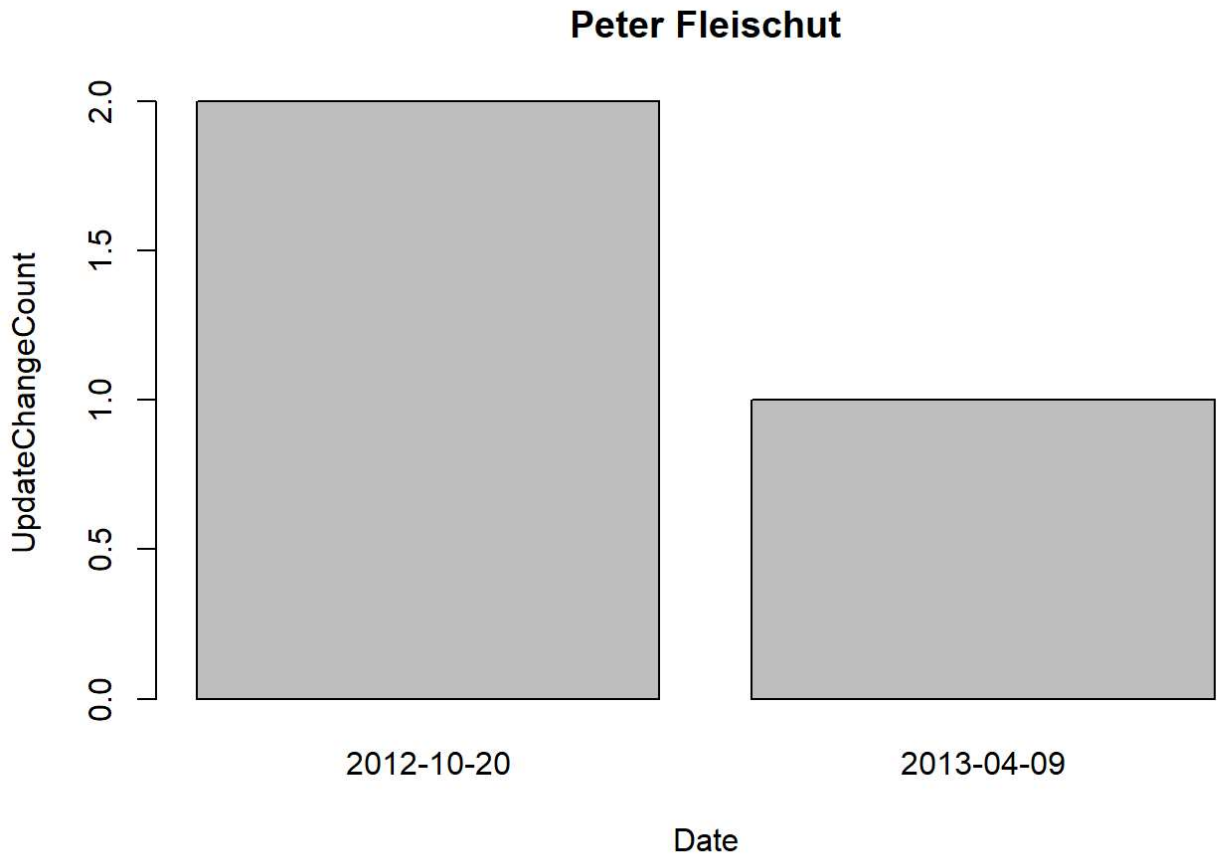

Hooman Kamel
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

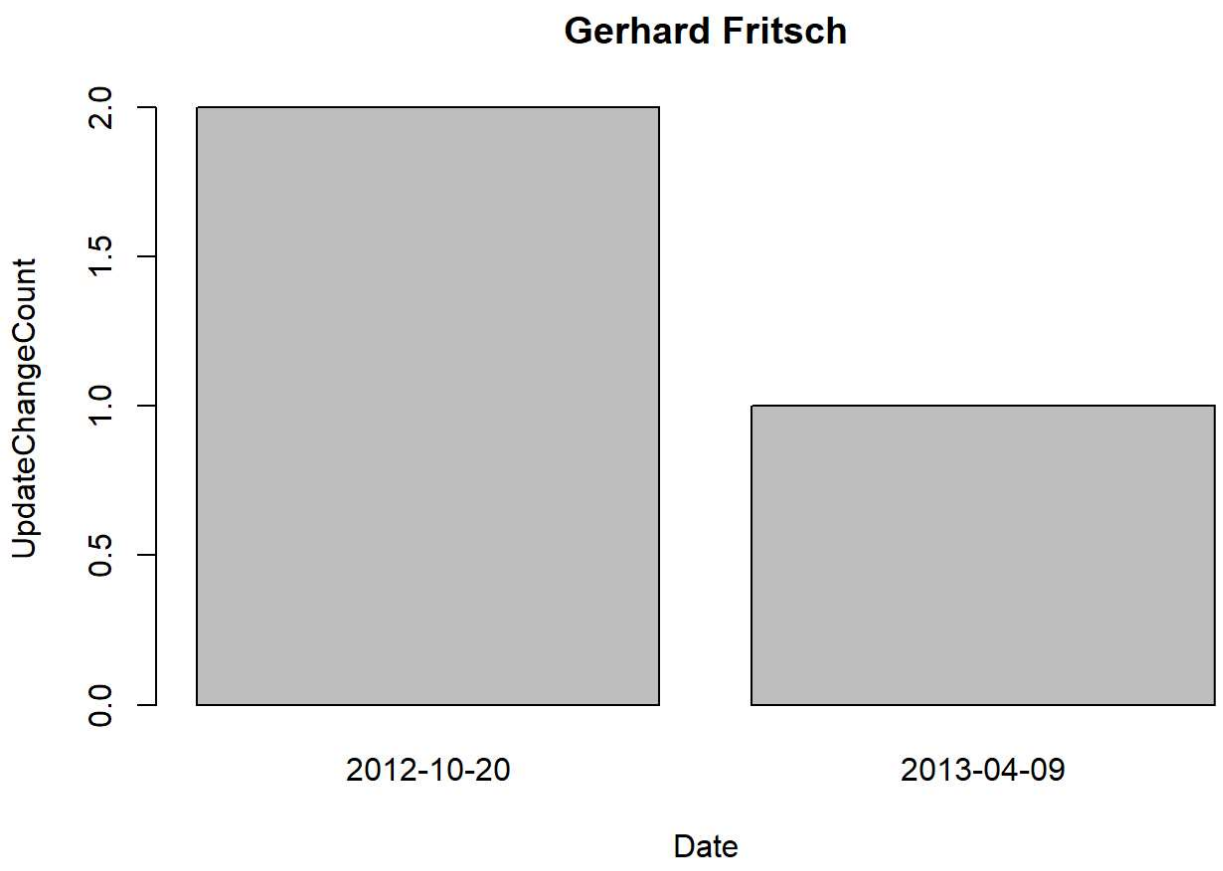

Dana Leifer
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

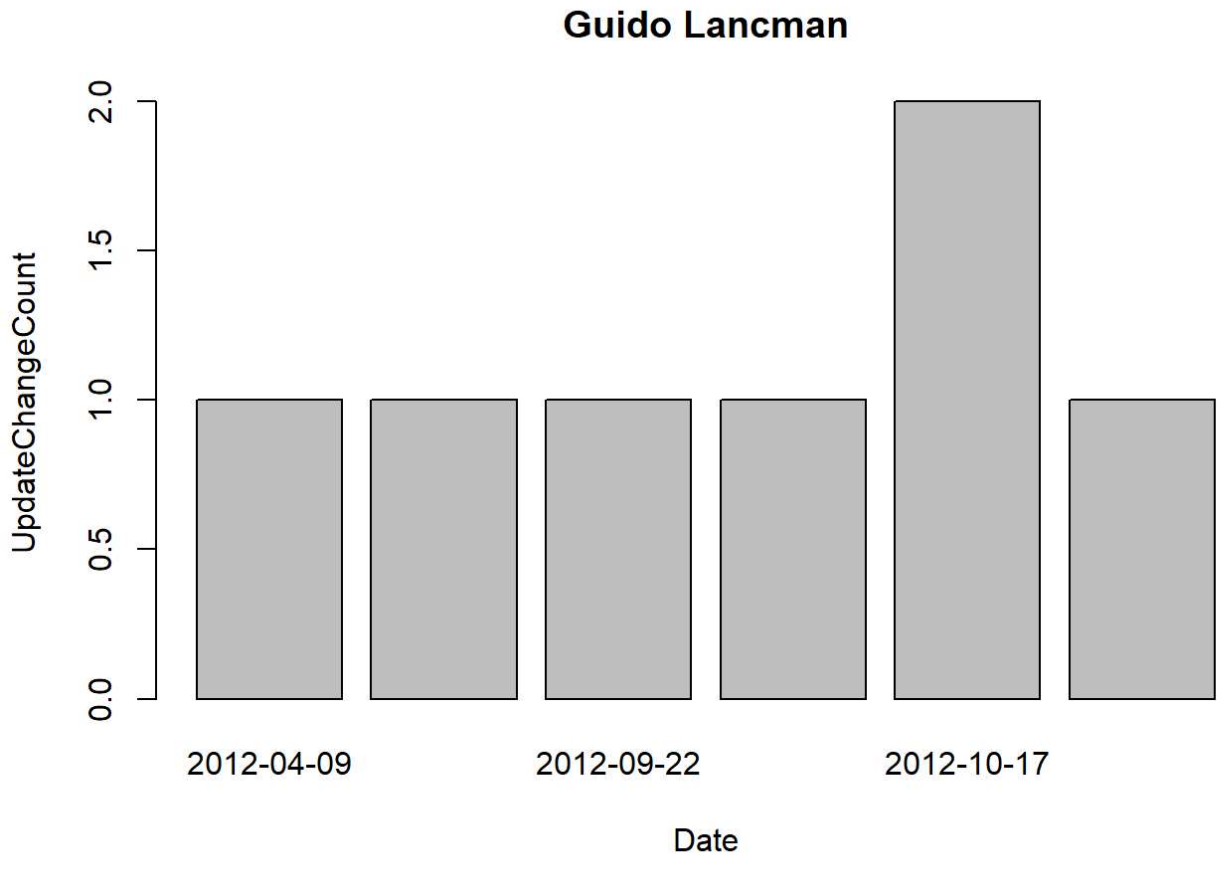

Pina C Sanelli
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-24
2013-01-04
Date

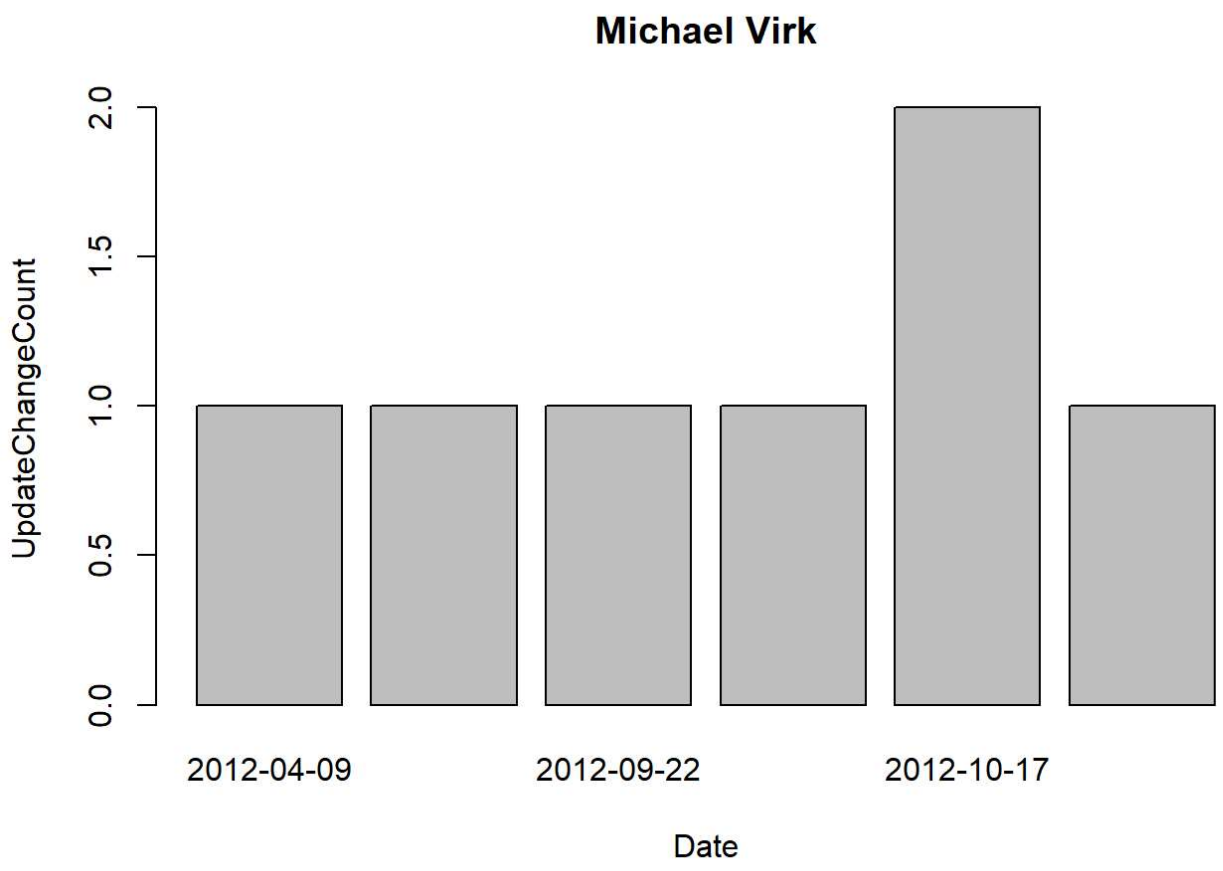

Ya-Lin Chiu
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-02-15
2011-11-23
2012-05-29
2013-01-03
2013-07-09
Date

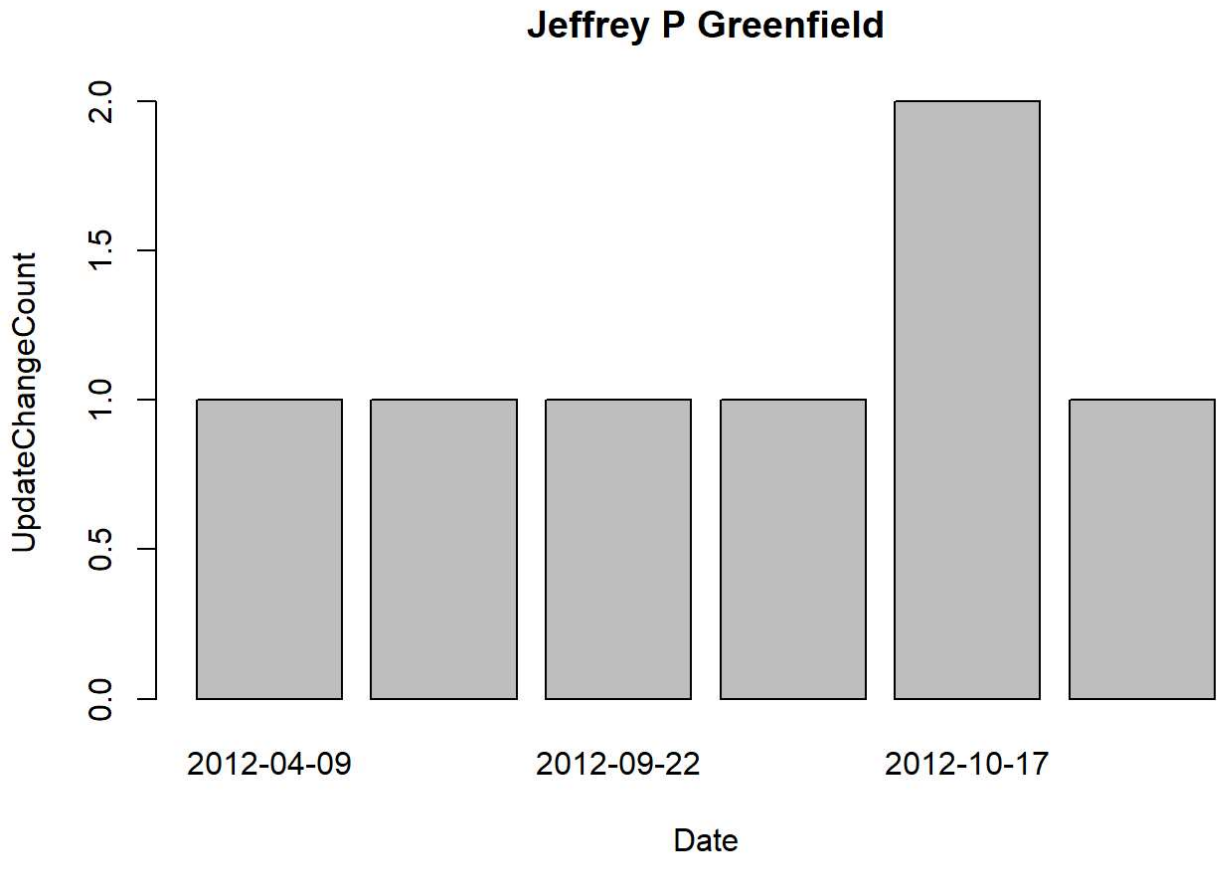

Xuming Sun
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-05-29
2012-10-20
2013-04-09
2013-07-09
Date

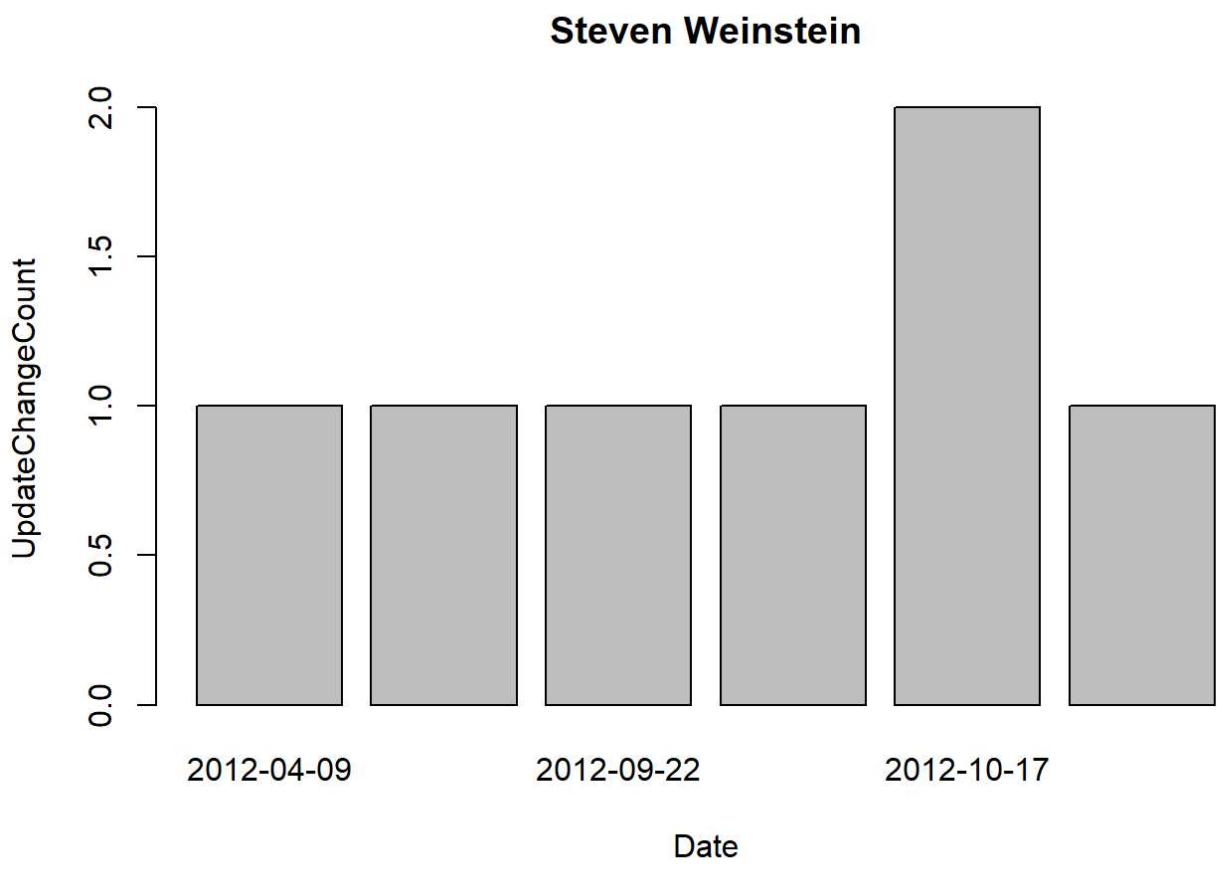

Peter Fleischut
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-20
2013-04-09
Date

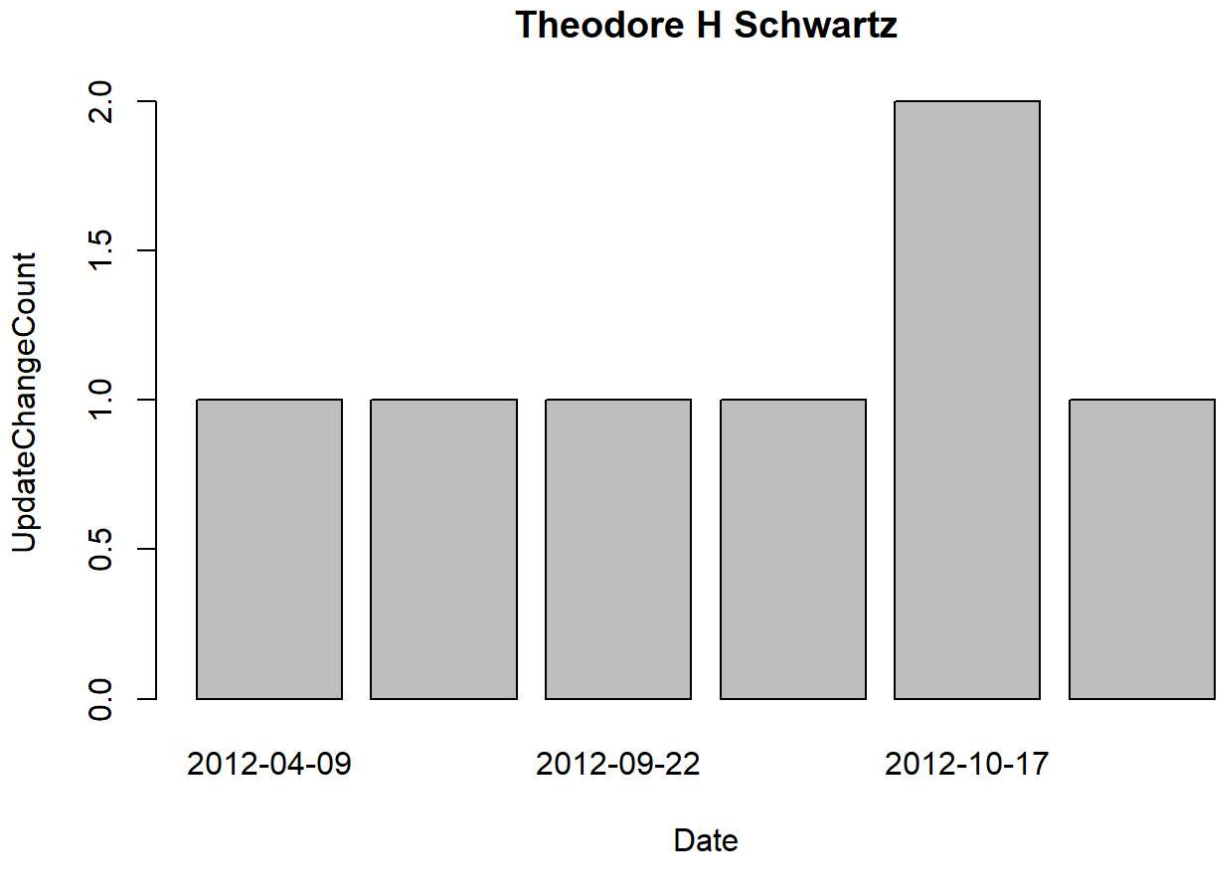

Gerhard Fritsch
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-10-20
2013-04-09
Date

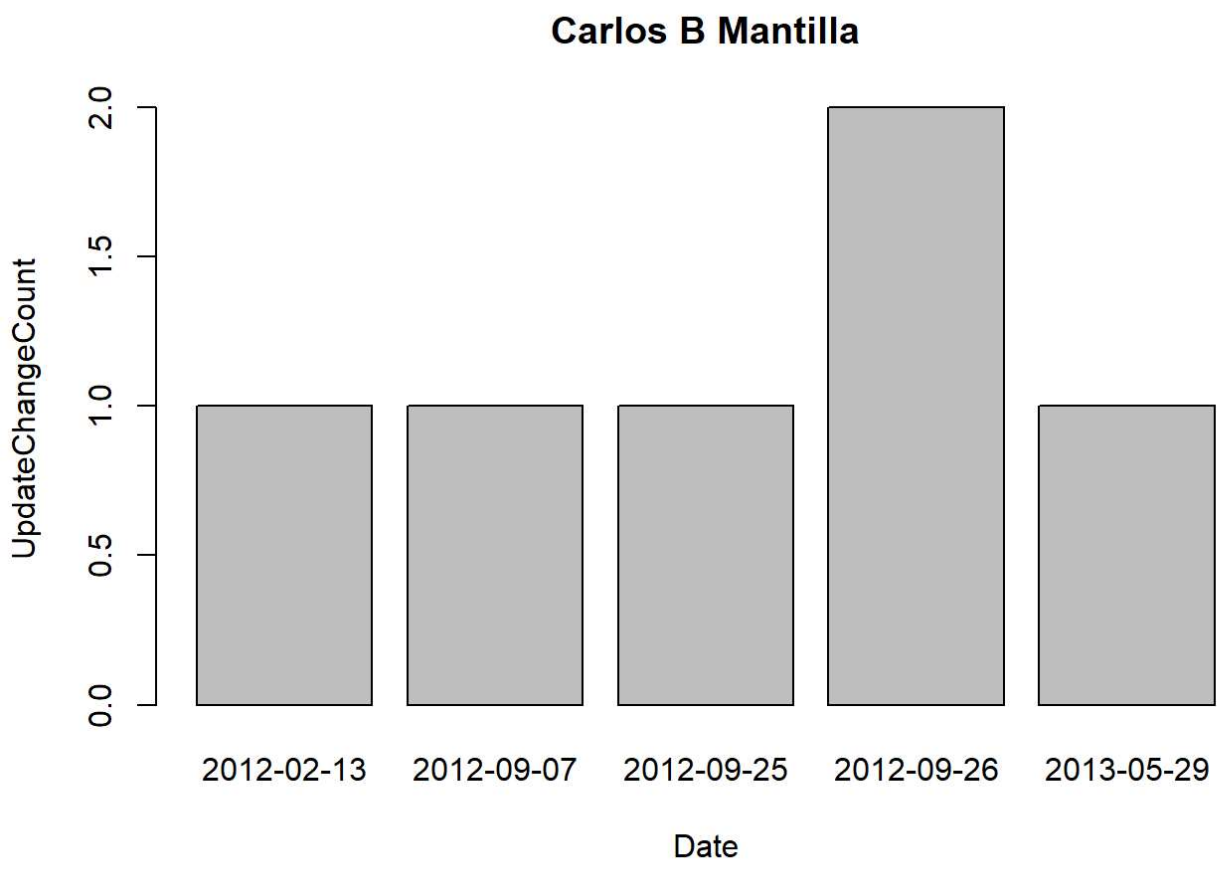

Guido Lancman
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-04-09
2012-09-22
2012-10-17
Date

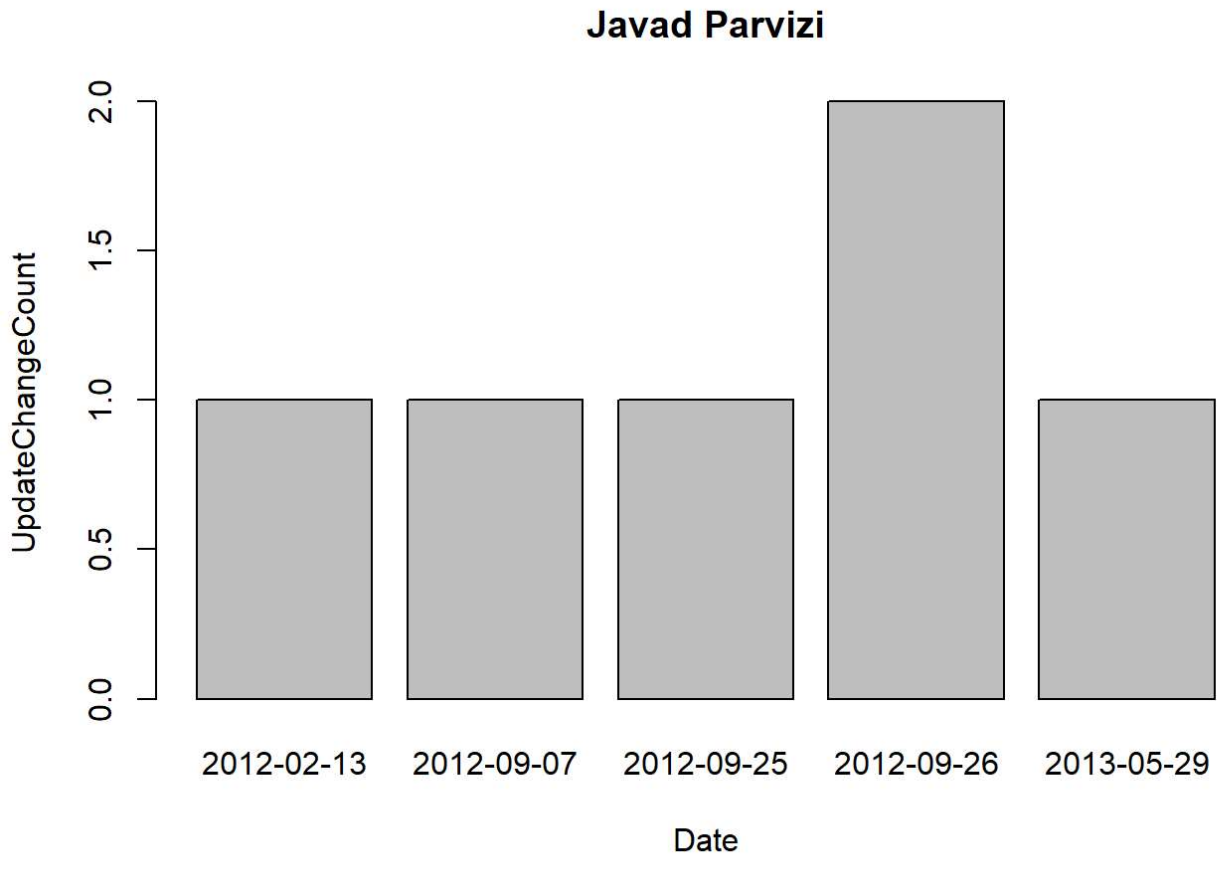

Michael Virk
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-04-09
2012-09-22
2012-10-17
Date

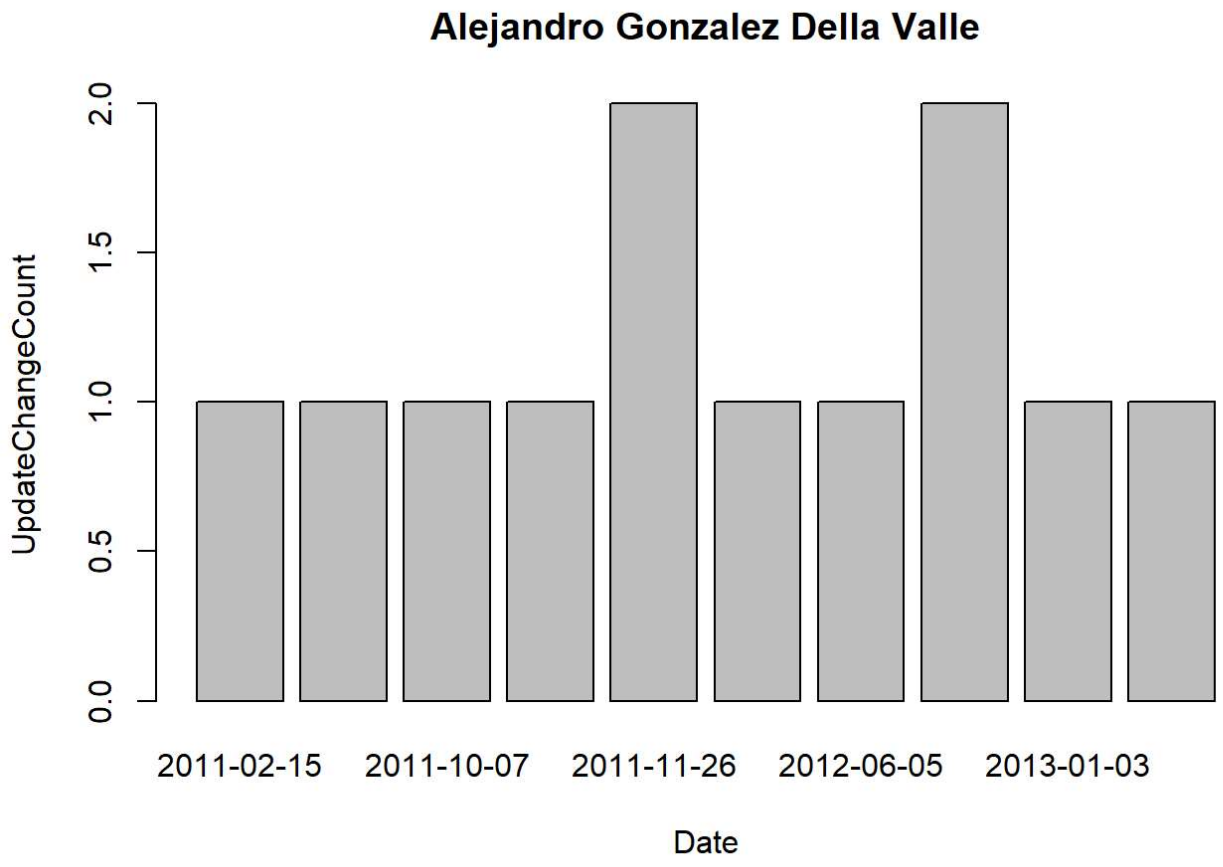

Jeffrey P Greenfield
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-04-09
2012-09-22
2012-10-17
Date

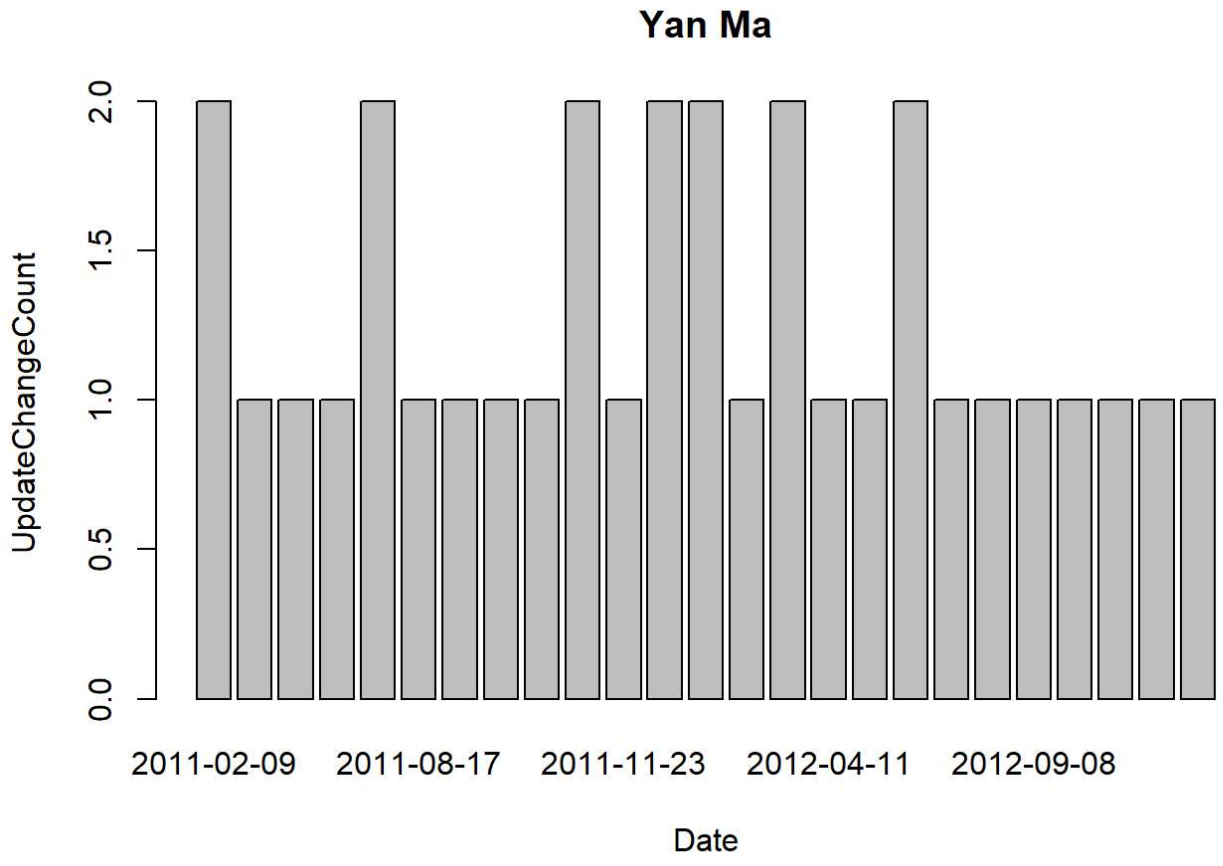

Steven Weinstein
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-04-09
2012-09-22
2012-10-17
Date

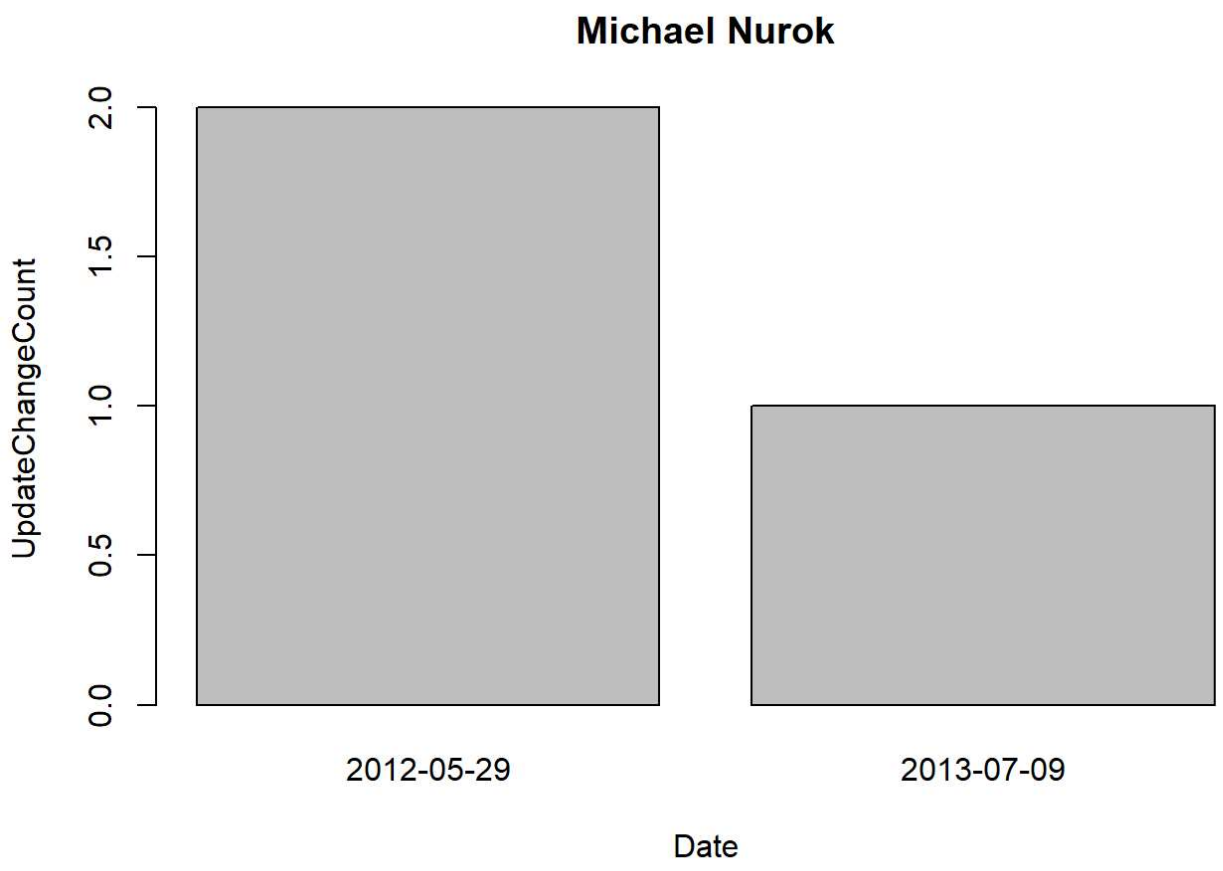

Theodore H Schwartz
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-04-09
2012-09-22
2012-10-17
Date

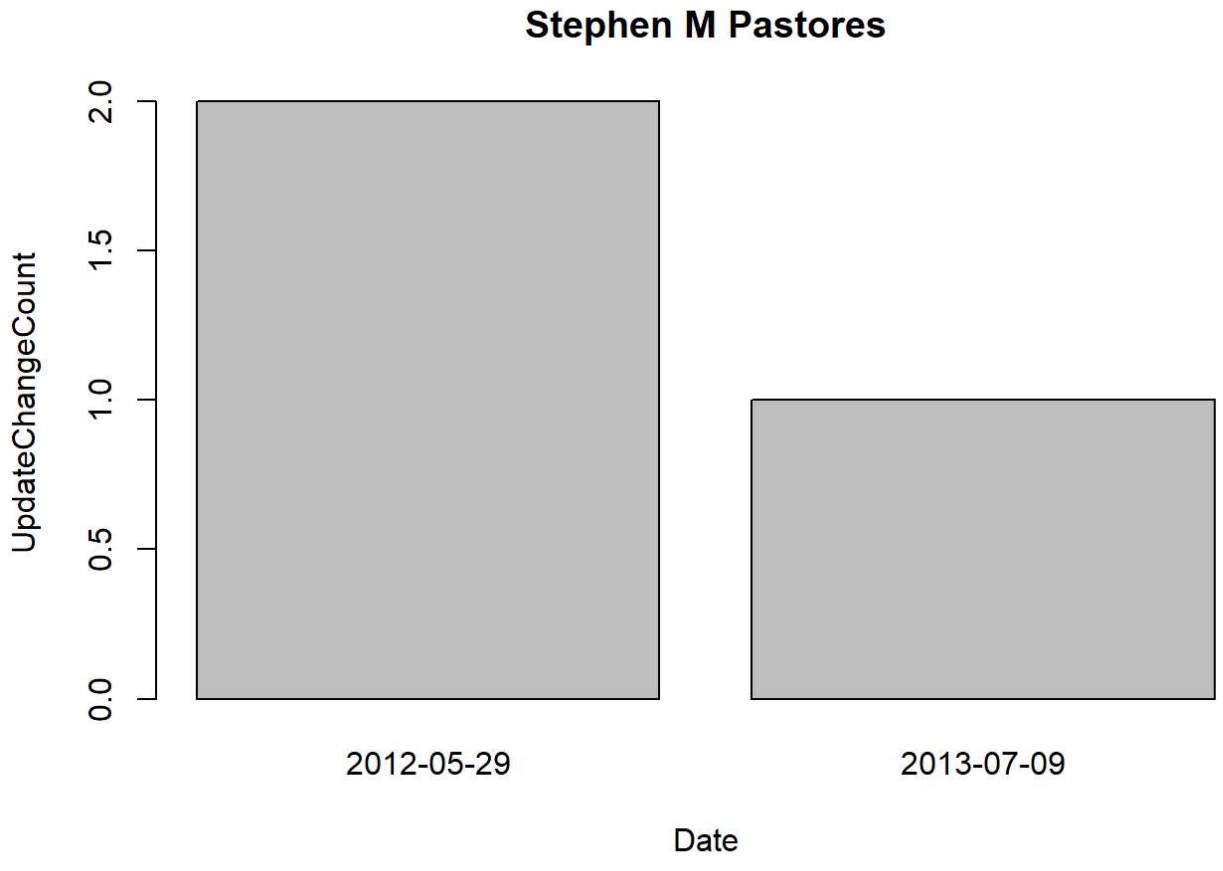

Carlos B Mantilla
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-02-13
2012-09-07
2012-09-25
2012-09-26
2013-05-29
Date

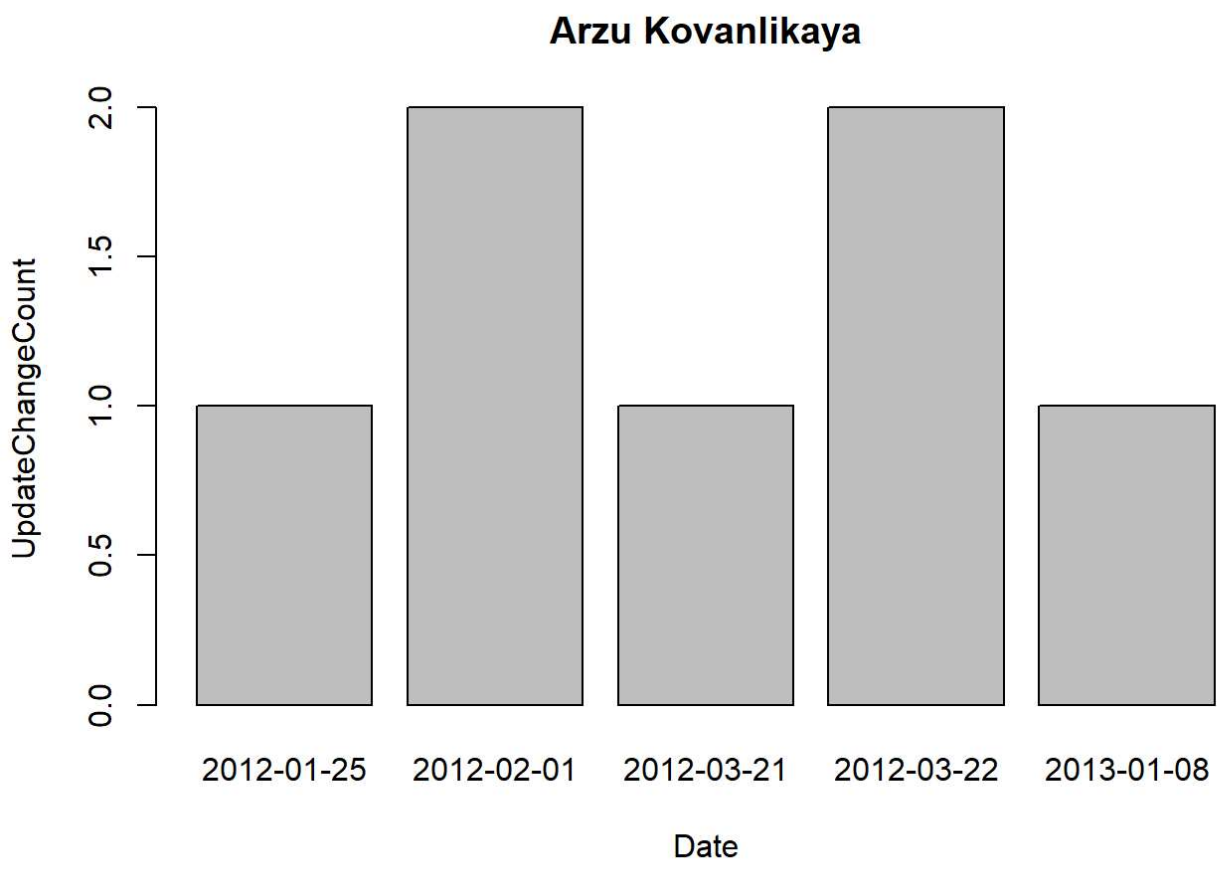

Javad Parvizi
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-02-13
2012-09-07
2012-09-25
2012-09-26
2013-05-29
Date

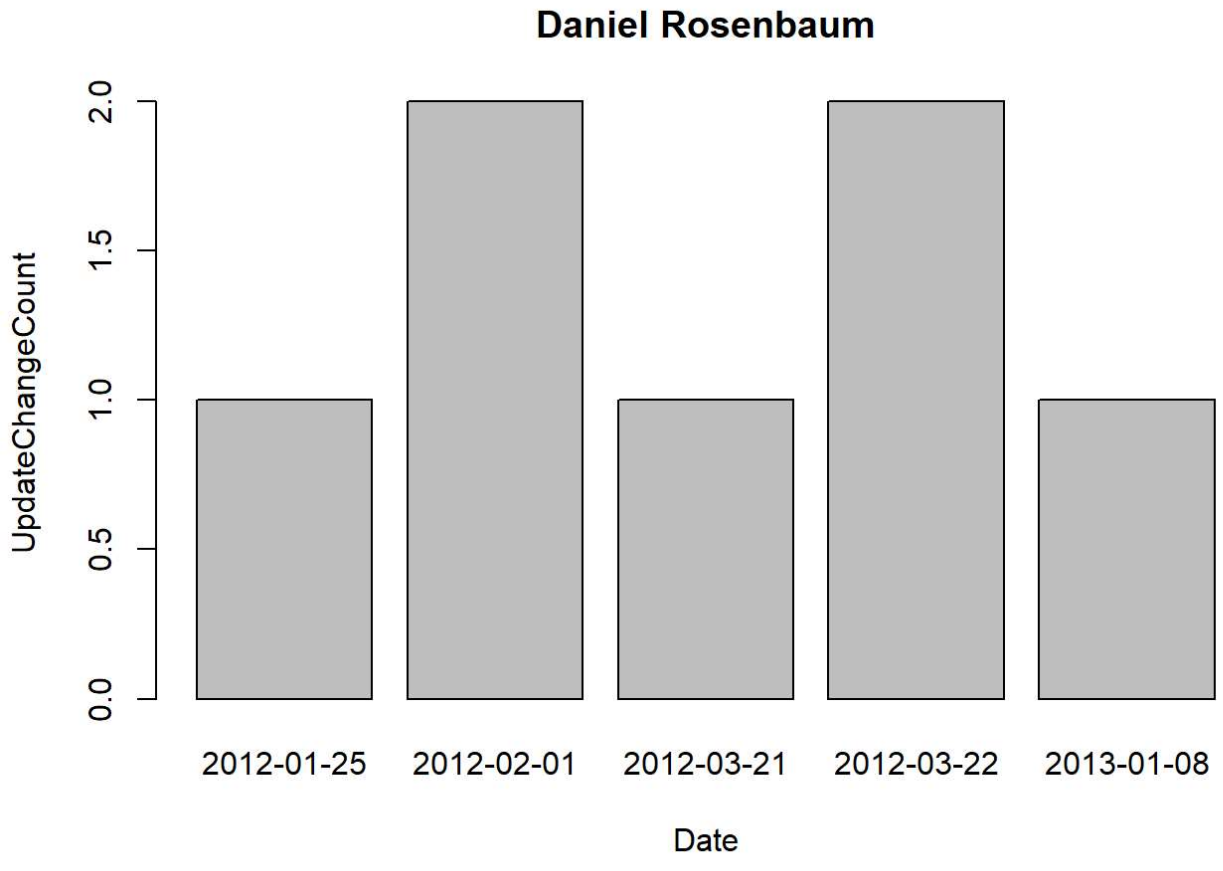

Alejandro Gonzalez Della Valle
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-02-15
2011-10-07
2011-11-26
2012-06-05
2013-01-03
Date

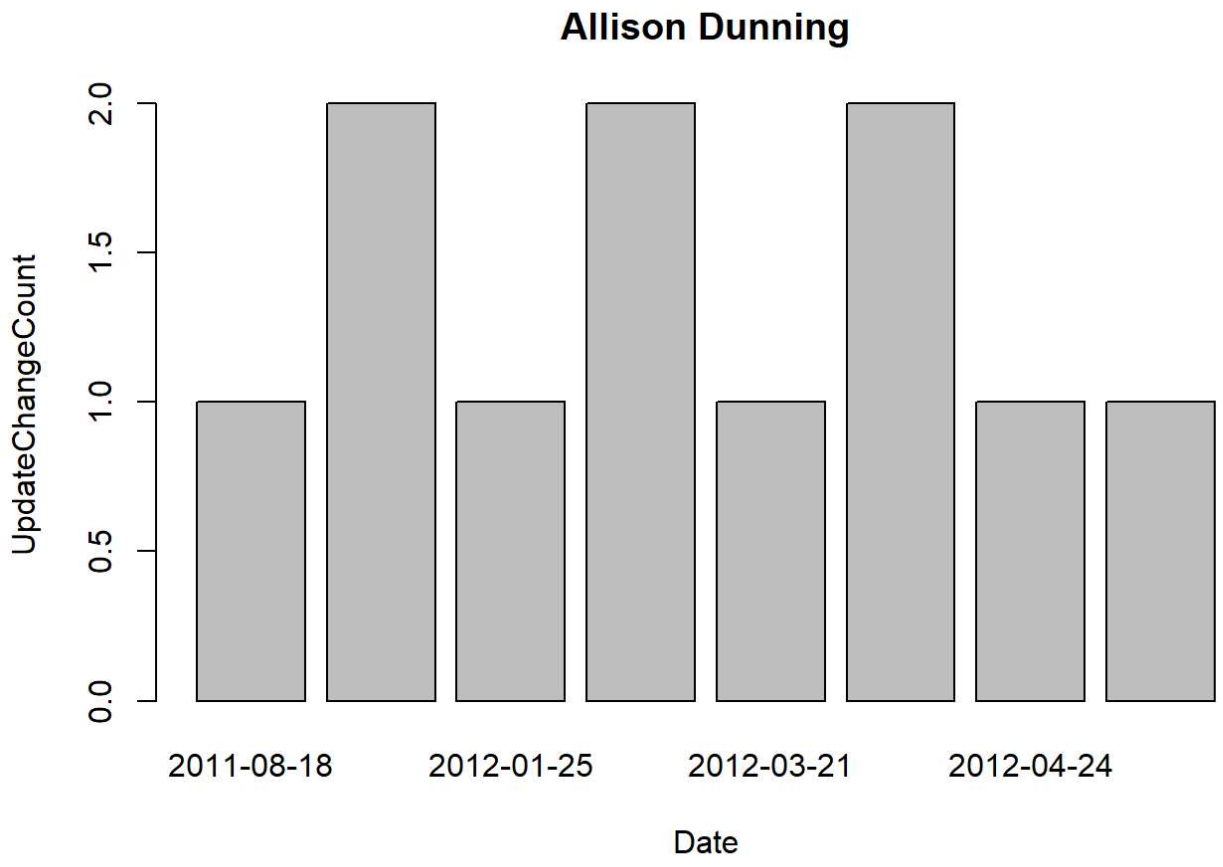

Yan Ma
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-02-09
2011-08-17
2011-11-23
2012-04-11
2012-09-08
Date

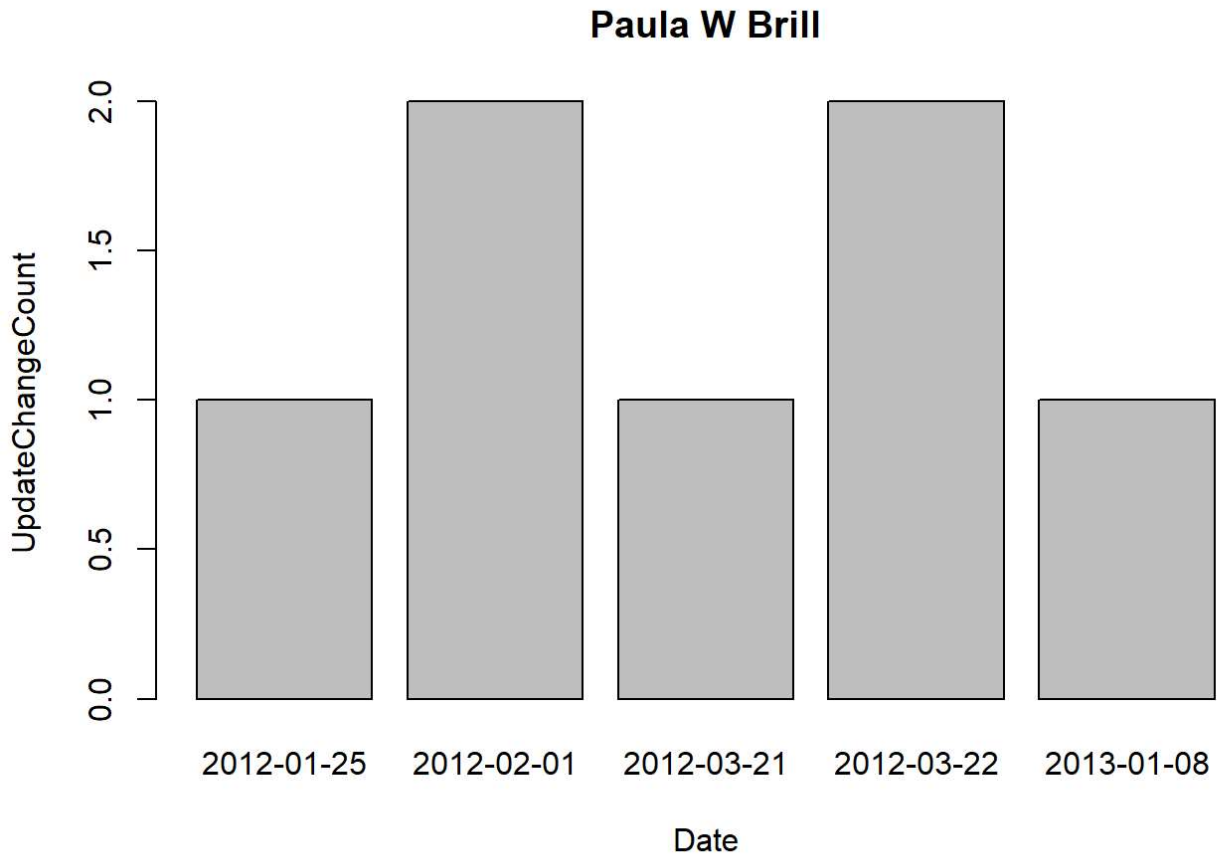

Michael Nurok
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-05-29
2013-07-09
Date

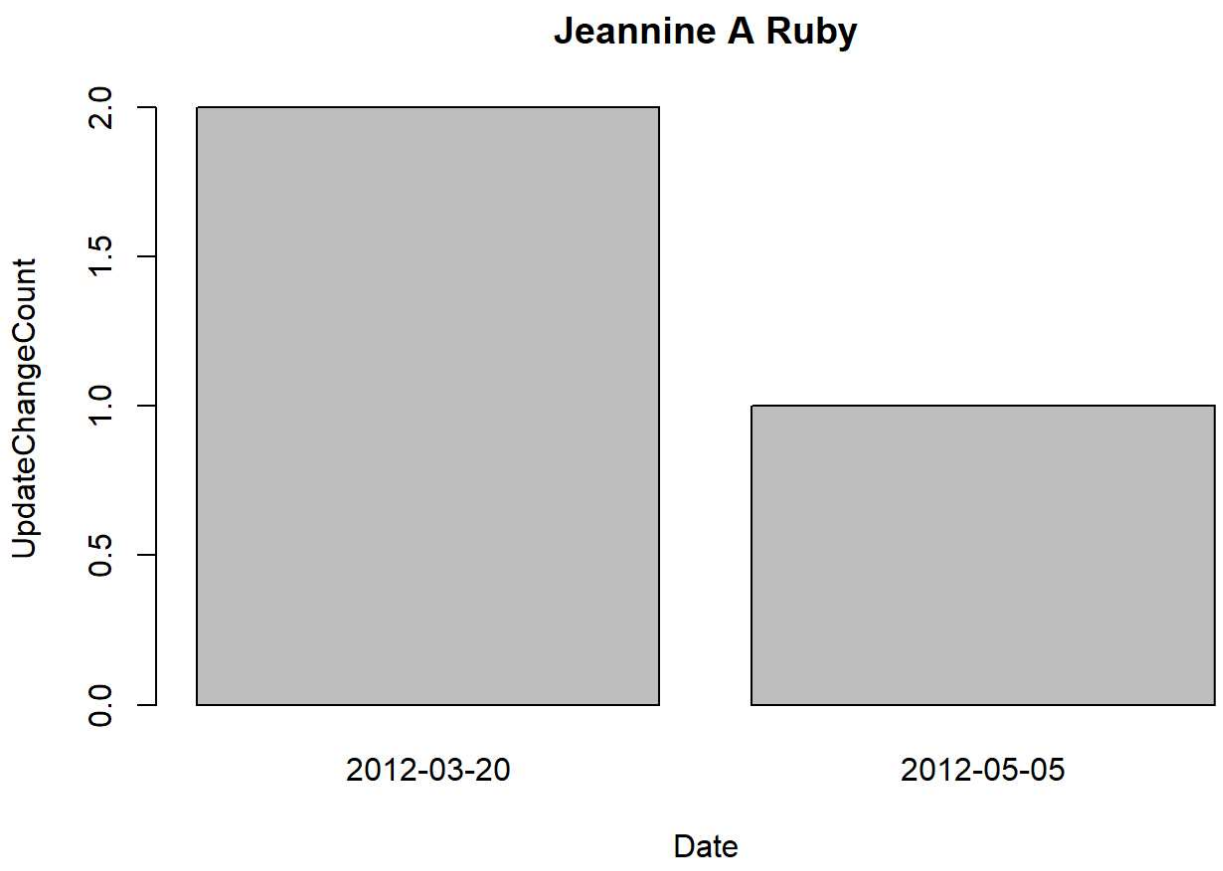

Stephen M Pastores
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-05-29
2013-07-09
Date

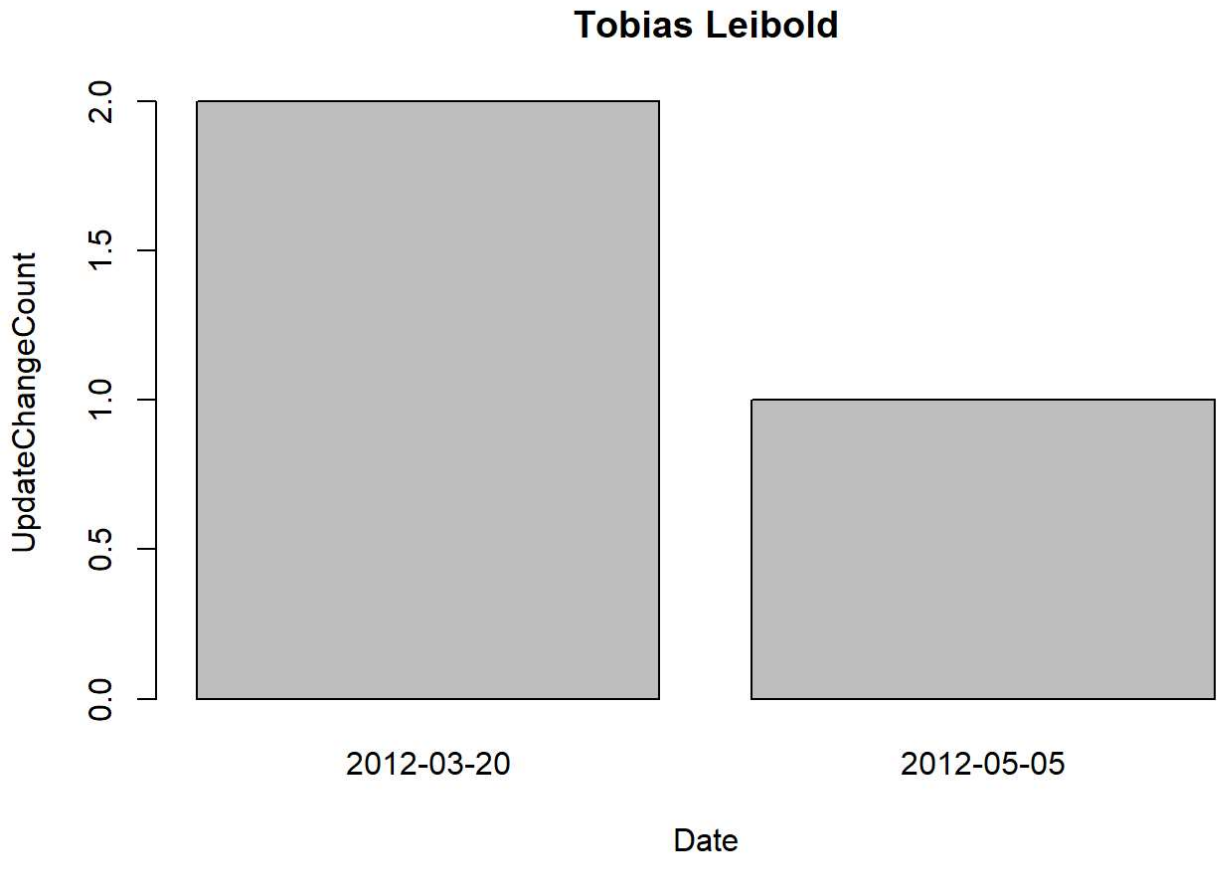

Arzu Kovanlikaya
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-01-25
2012-02-01
2012-03-21
2012-03-22
2013-01-08
Date

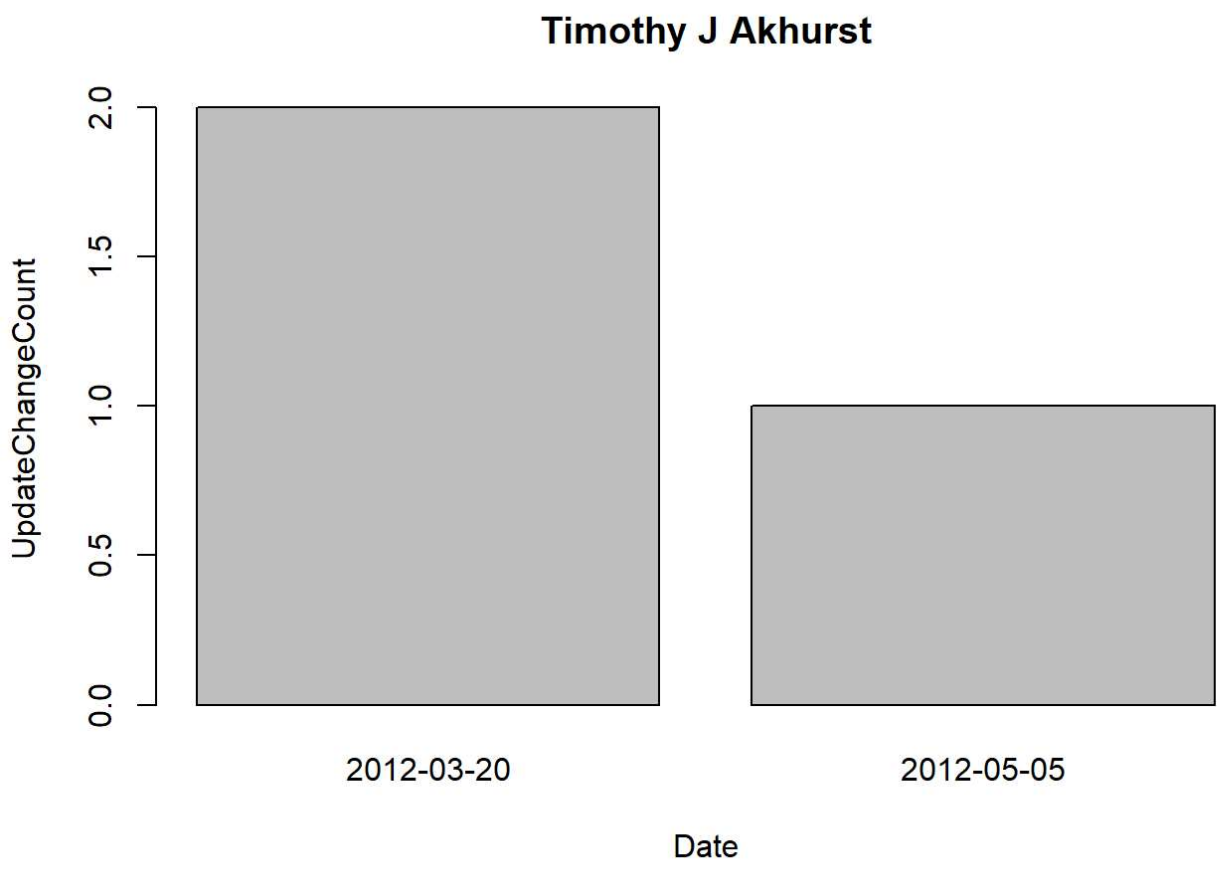

Daniel Rosenbaum
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-01-25
2012-02-01
2012-03-21
2012-03-22
2013-01-08
Date

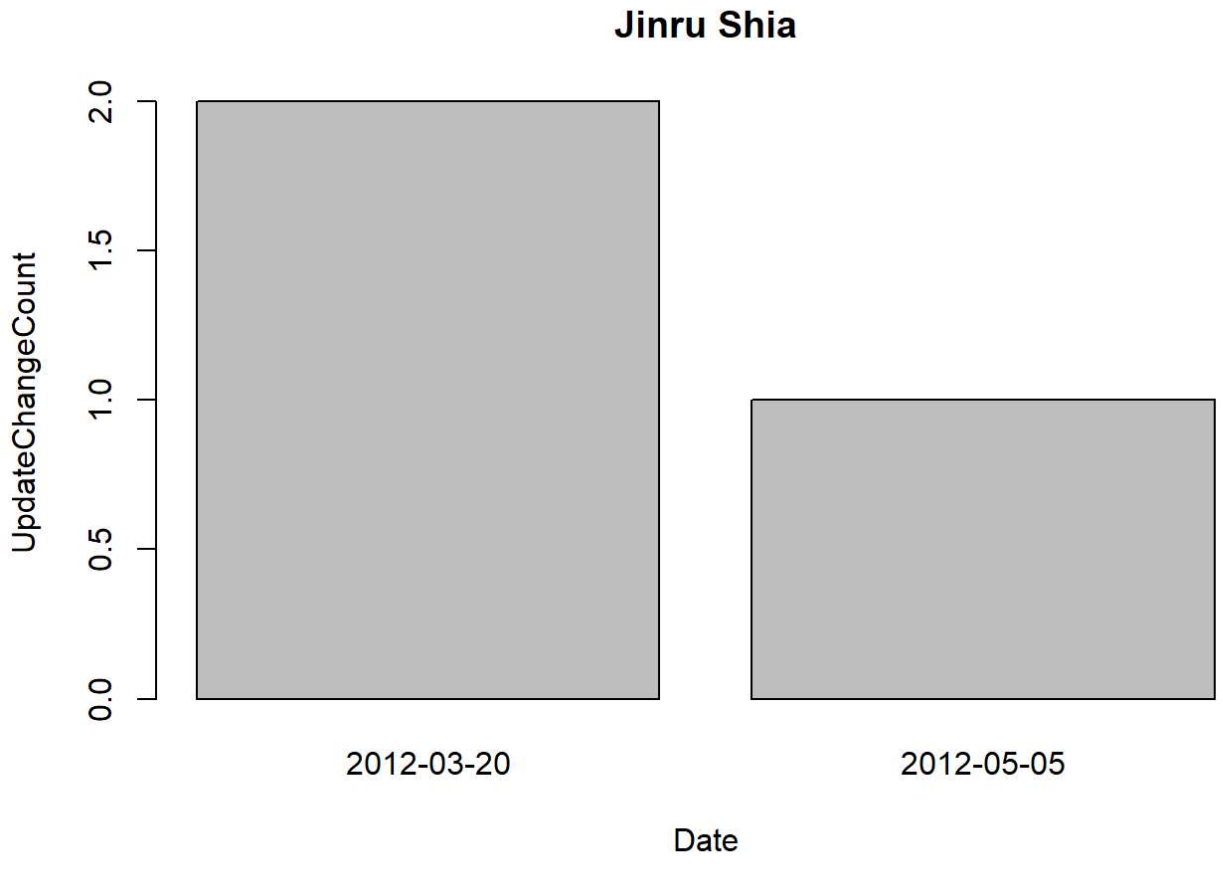

Allison Dunning
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2012-01-25
2012-03-21
2012-04-24
Date

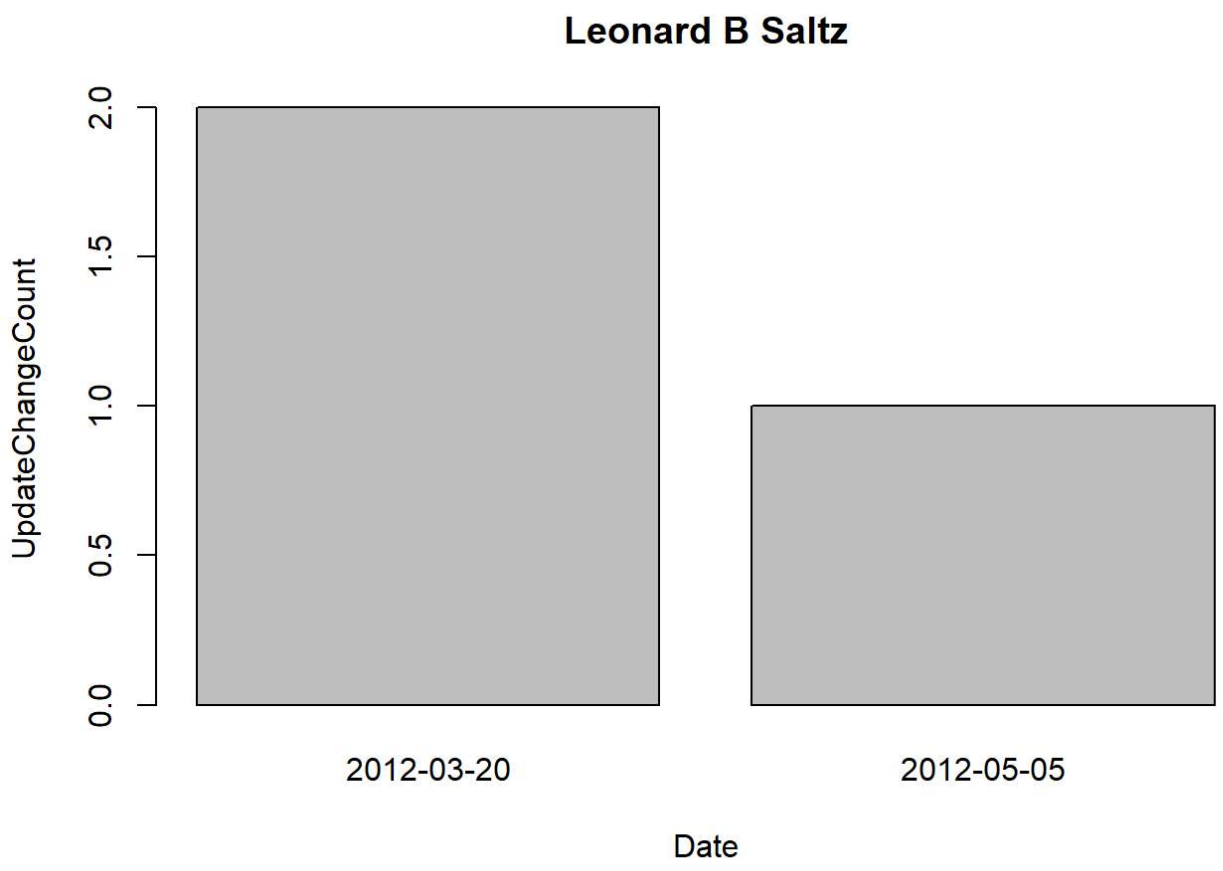

Paula W Brill
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-01-25
2012-02-01
2012-03-21
2012-03-22
2013-01-08
Date

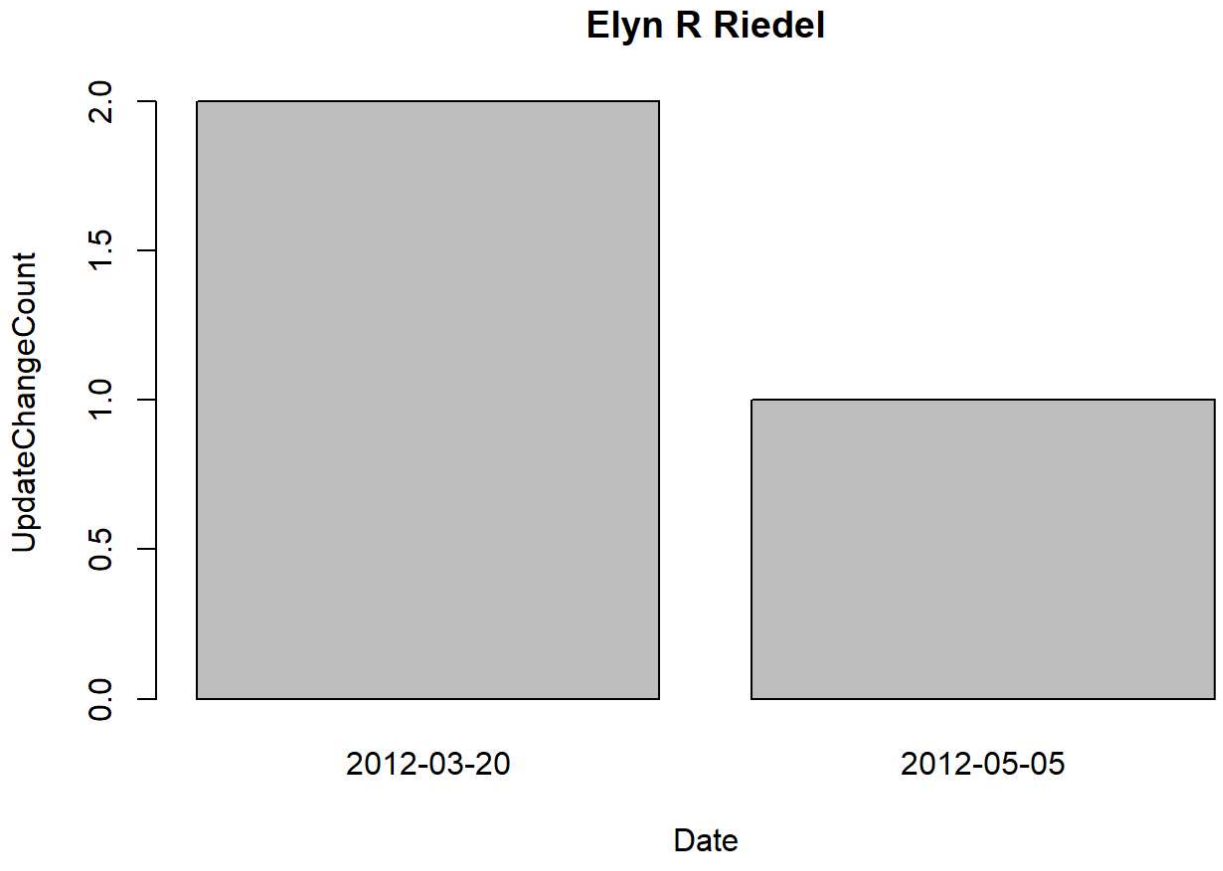

Jeannine A Ruby
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

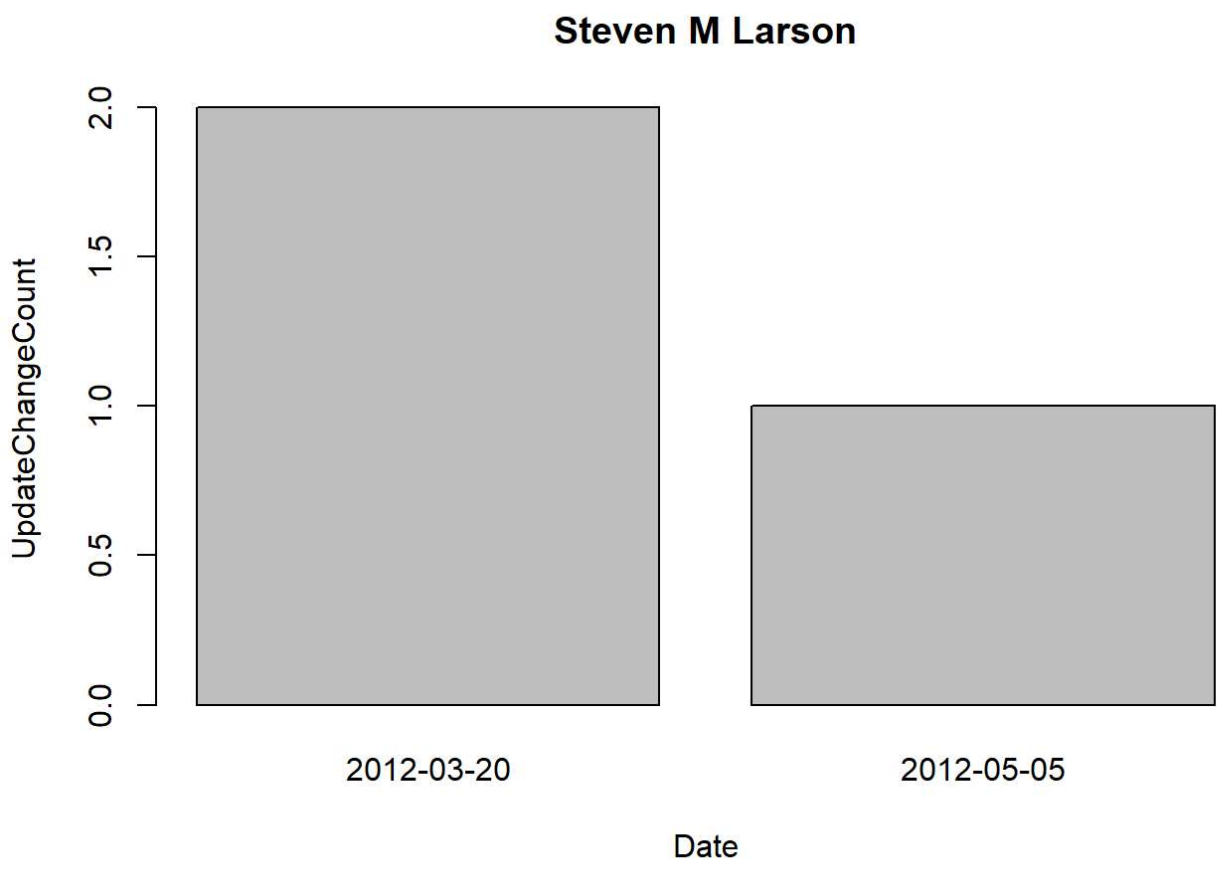

Tobias Leibold
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

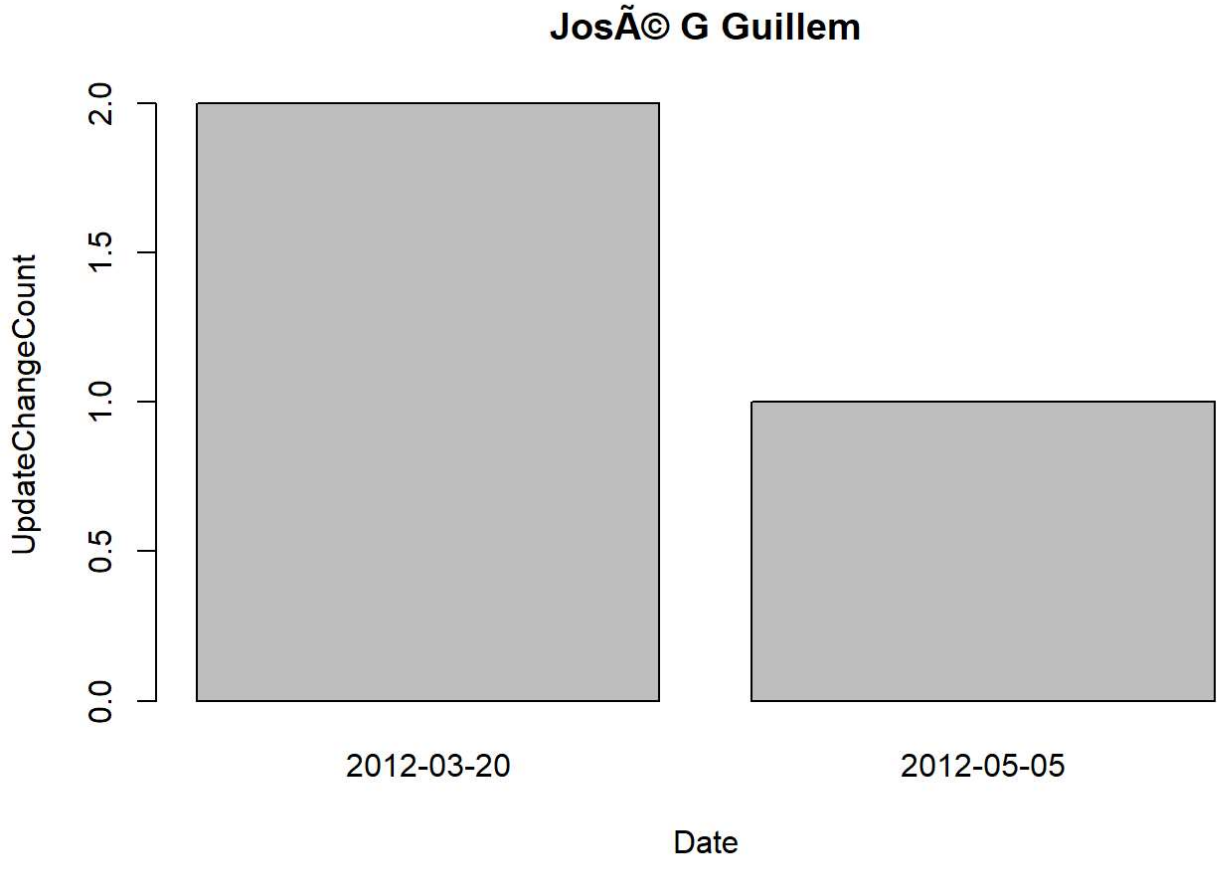

Timothy J Akhurst
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

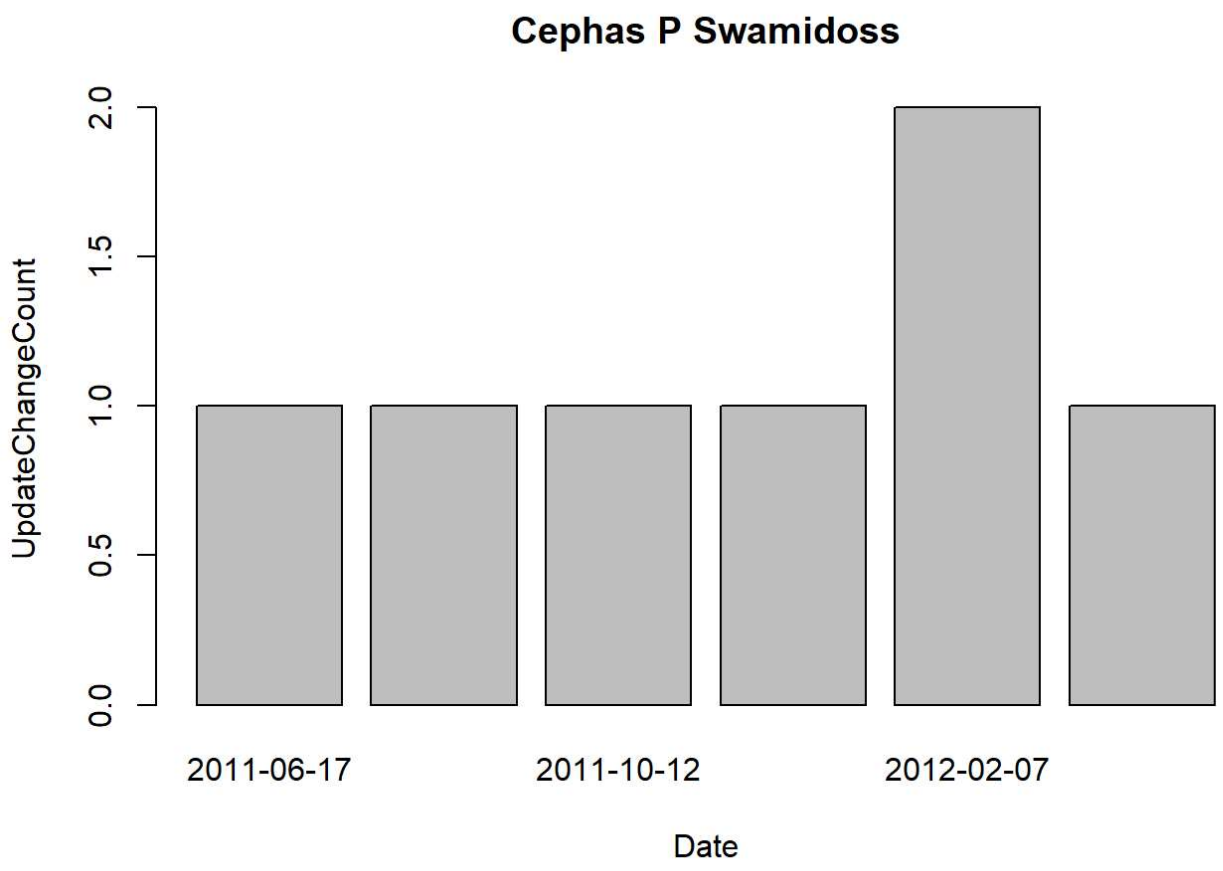

Jinru Shia
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

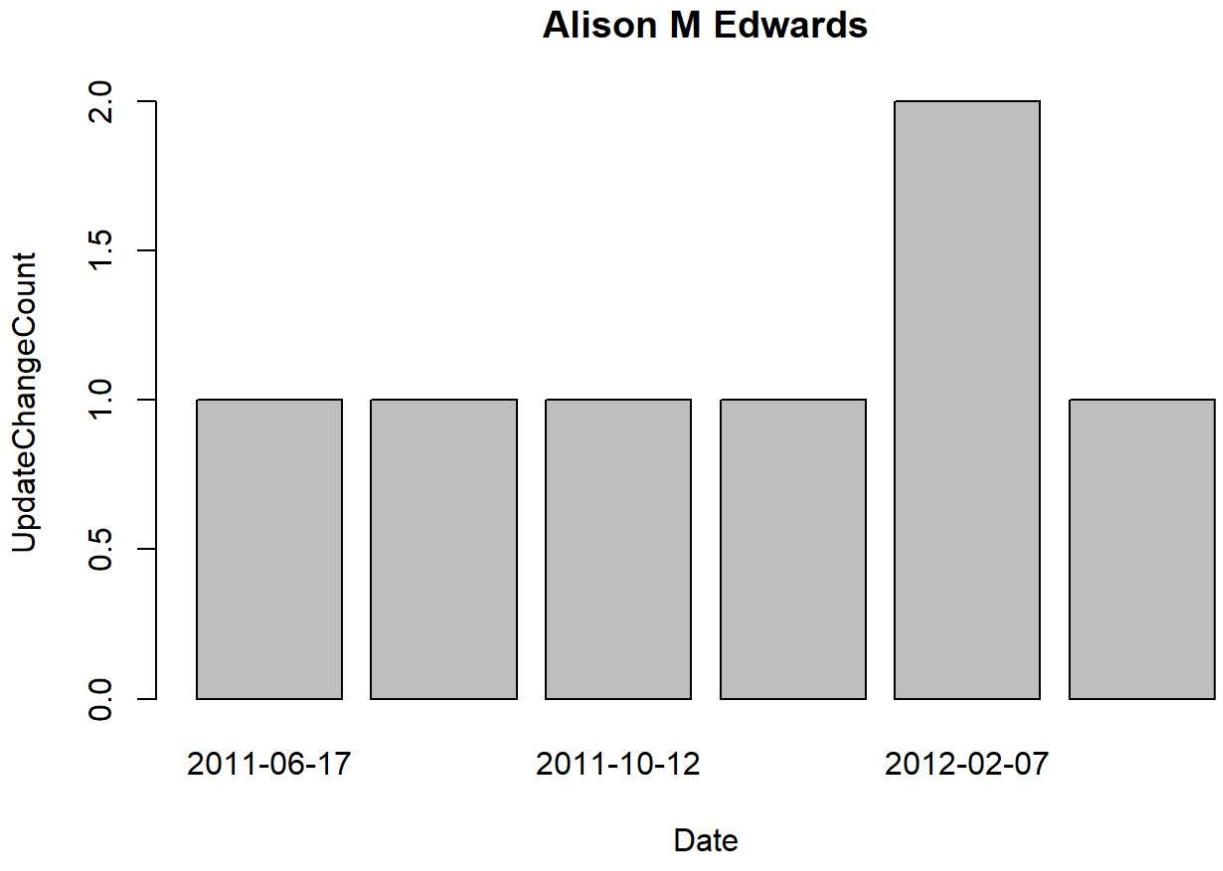

Leonard B Saltz
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

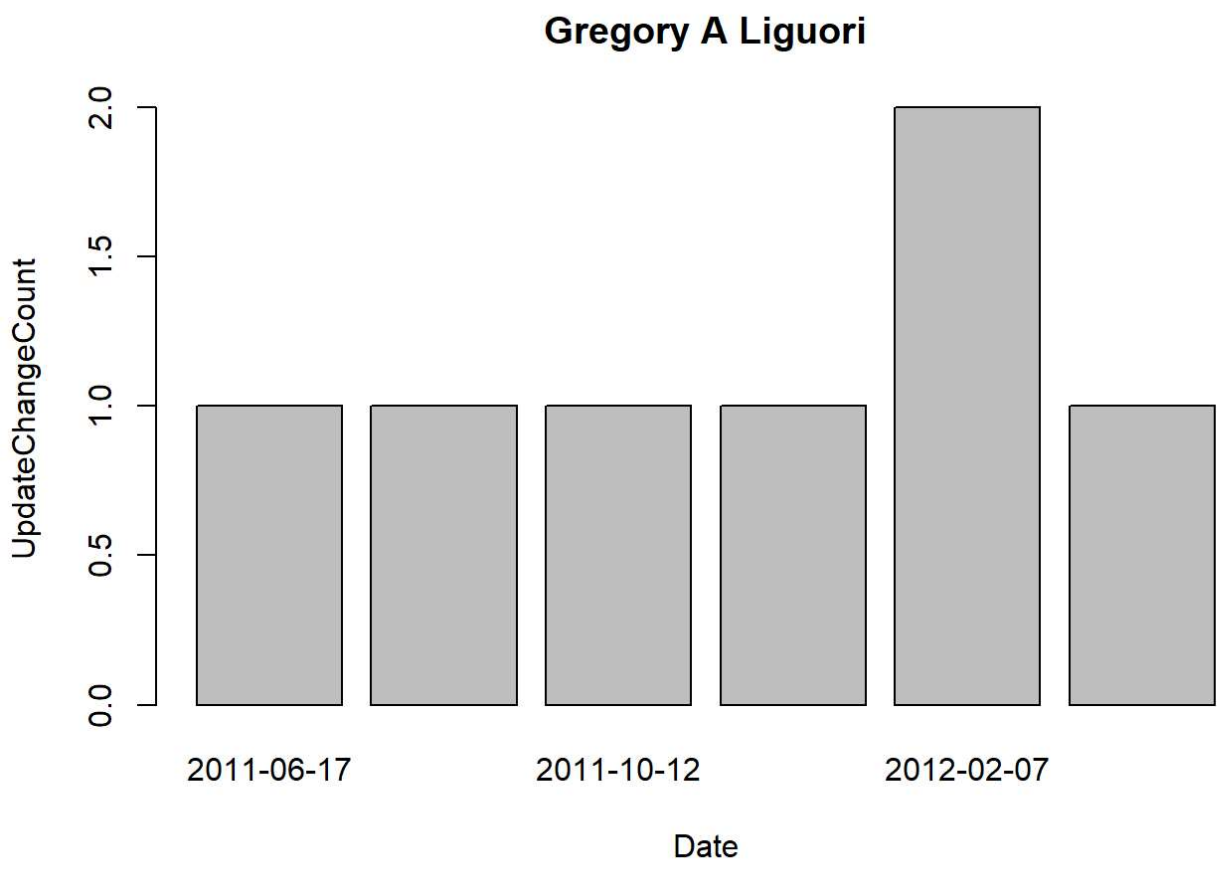

Elyn R Riedel
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

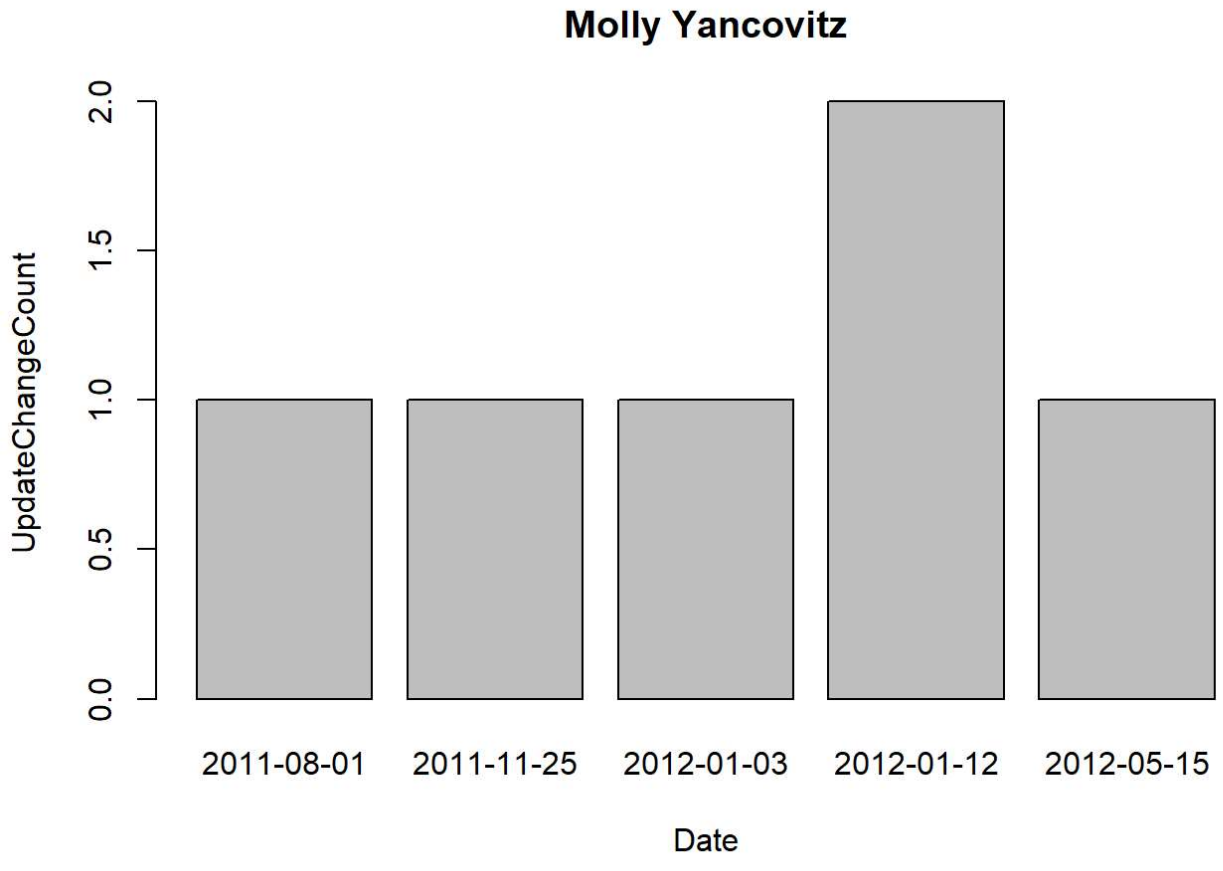

Steven M Larson
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

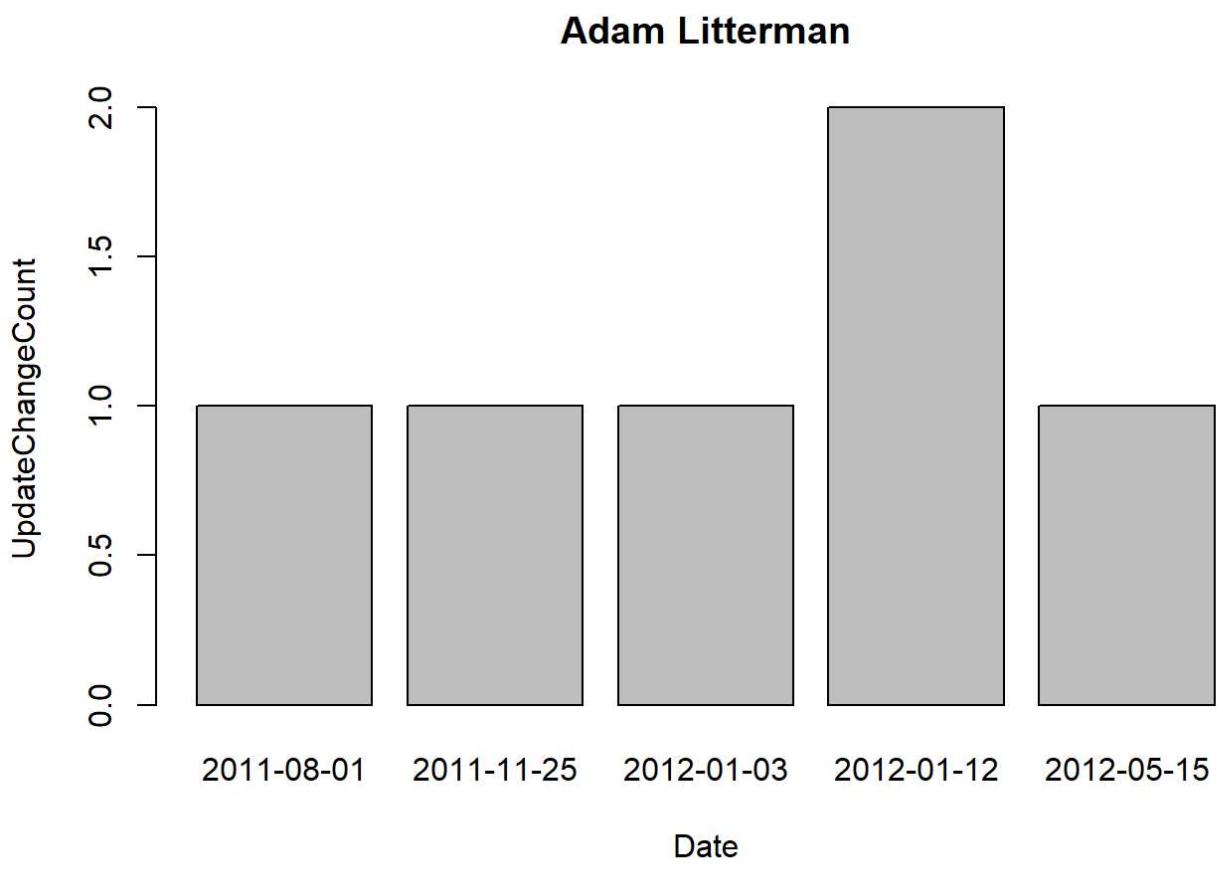

JosÃ© G Guillem
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2012-03-20
2012-05-05
Date

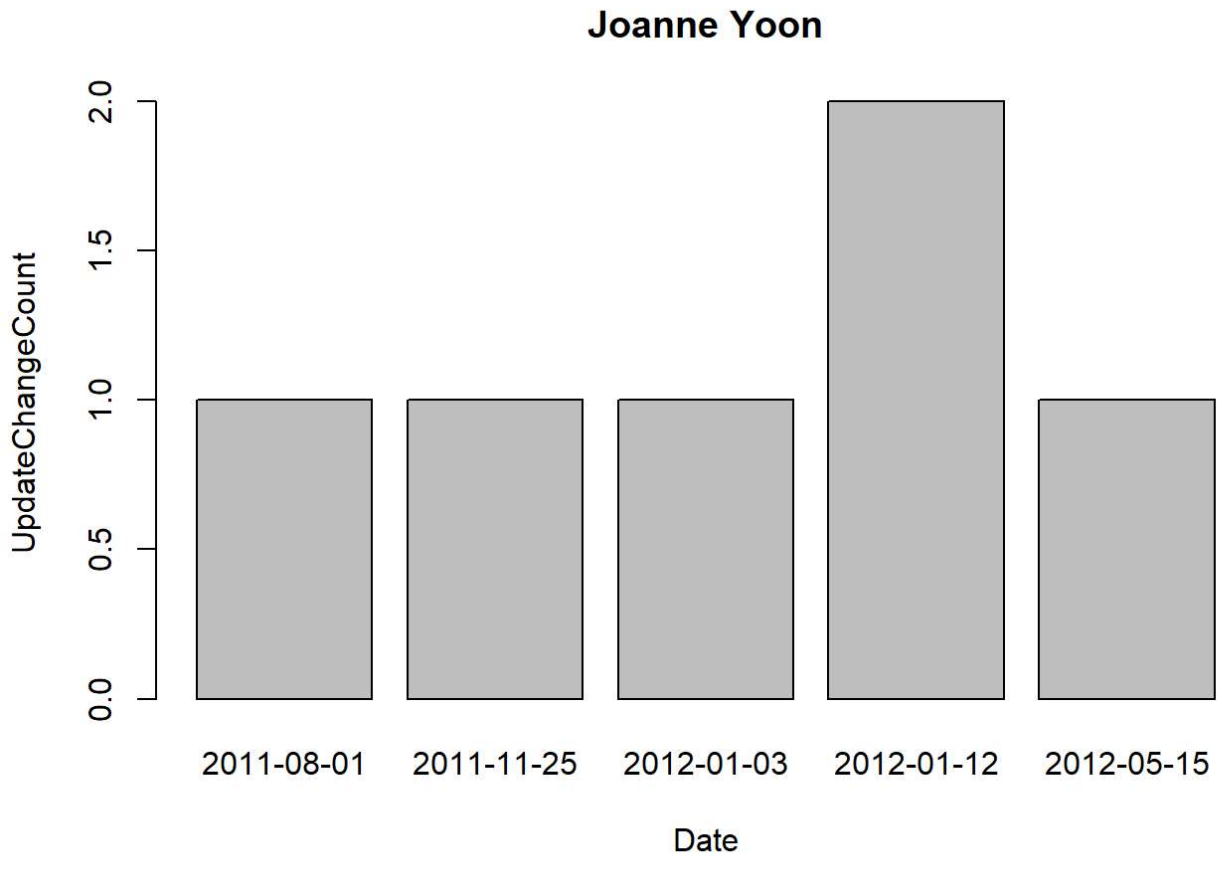

Cephas P Swamidoss
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-17
2011-10-12
2012-02-07
Date

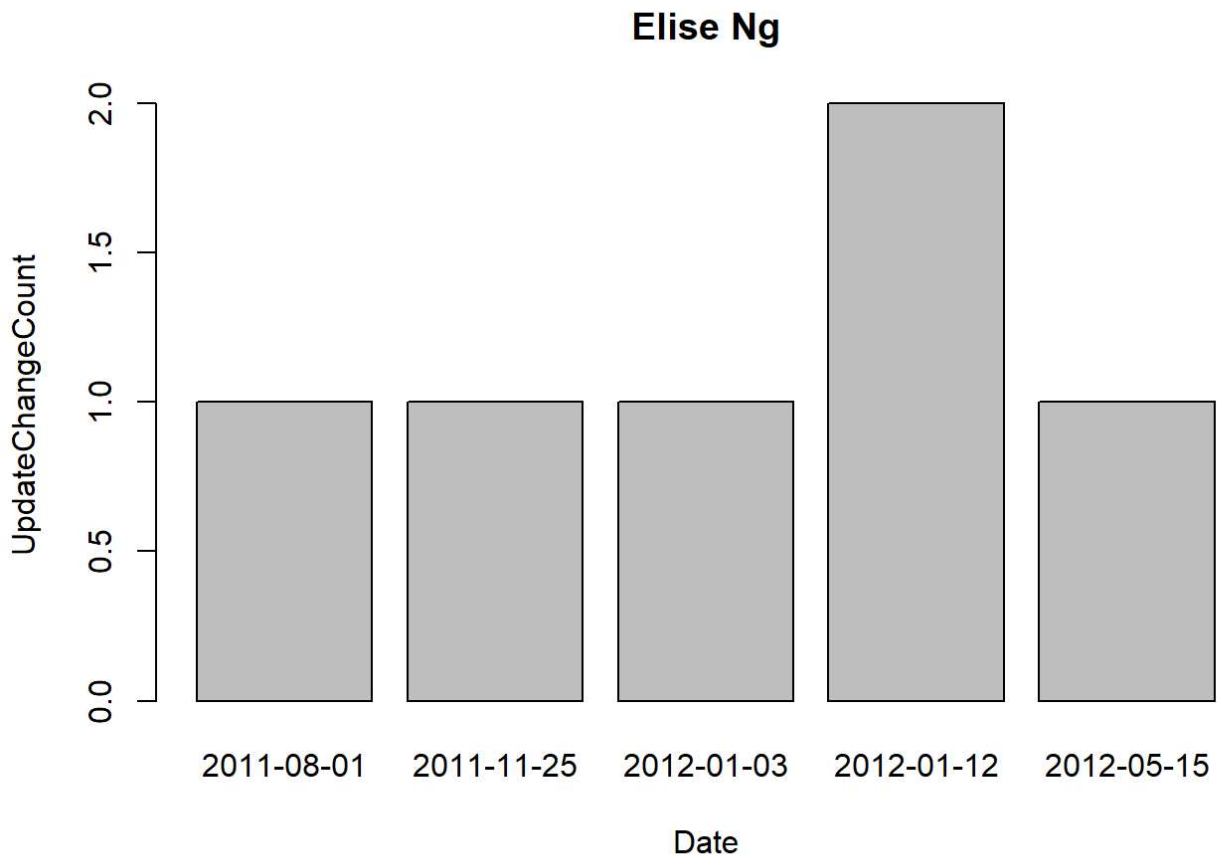

Alison M Edwards
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-17
2011-10-12
2012-02-07
Date

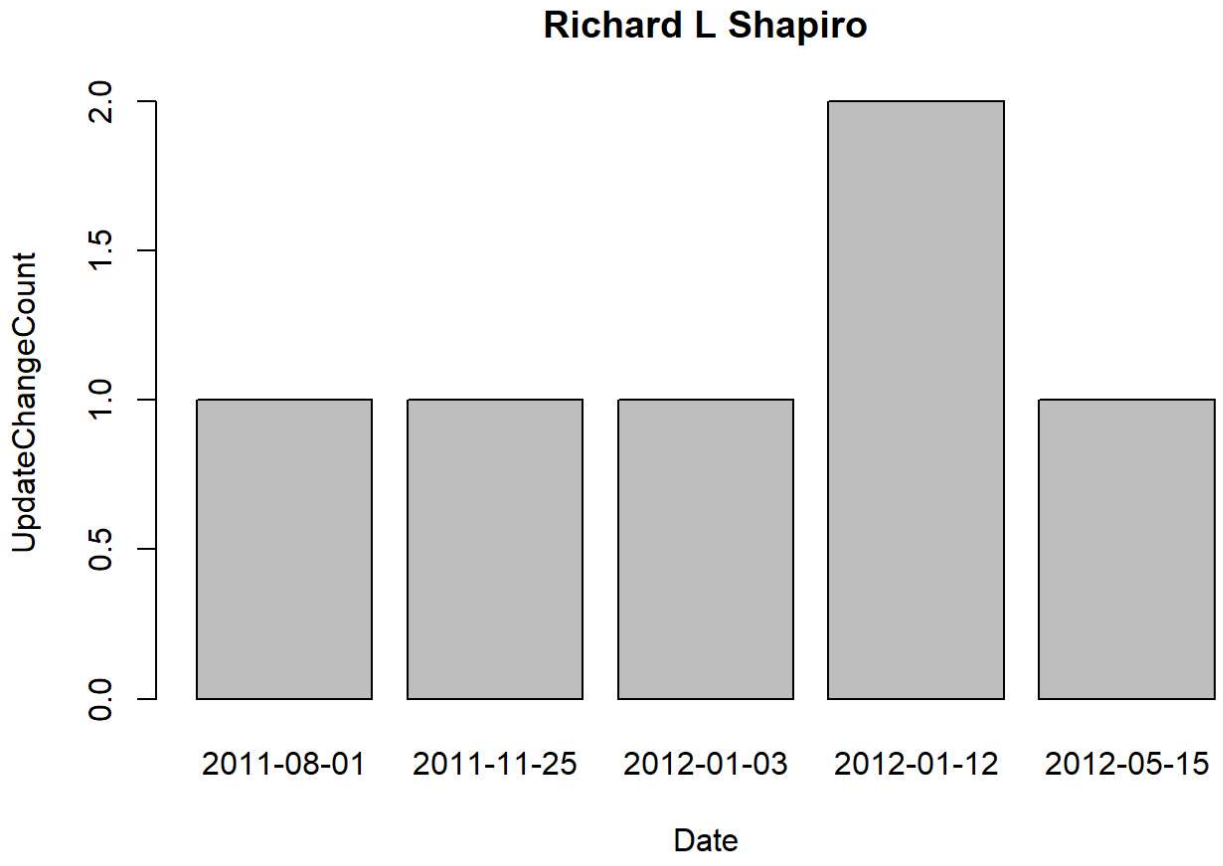

Gregory A Liguori
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-17
2011-10-12
2012-02-07
Date

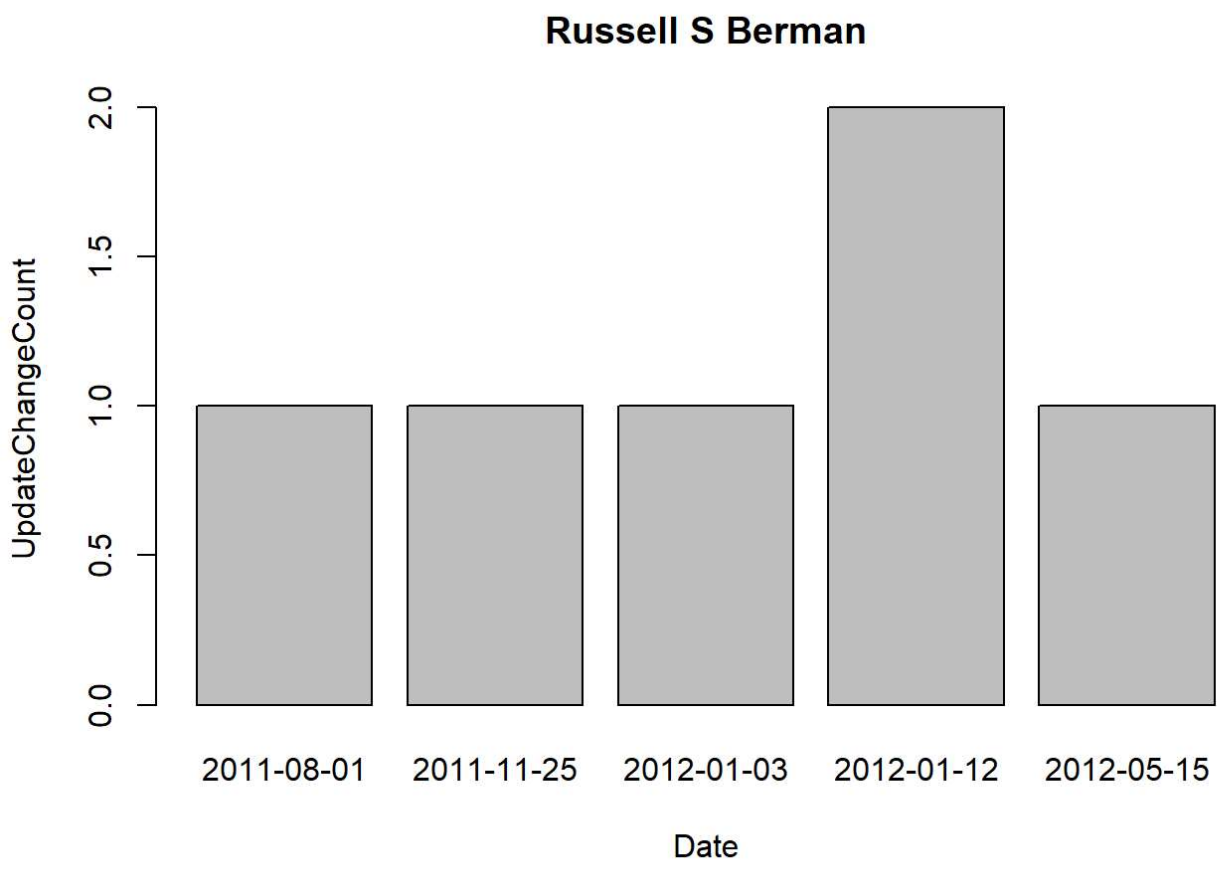

Molly Yancovitz
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

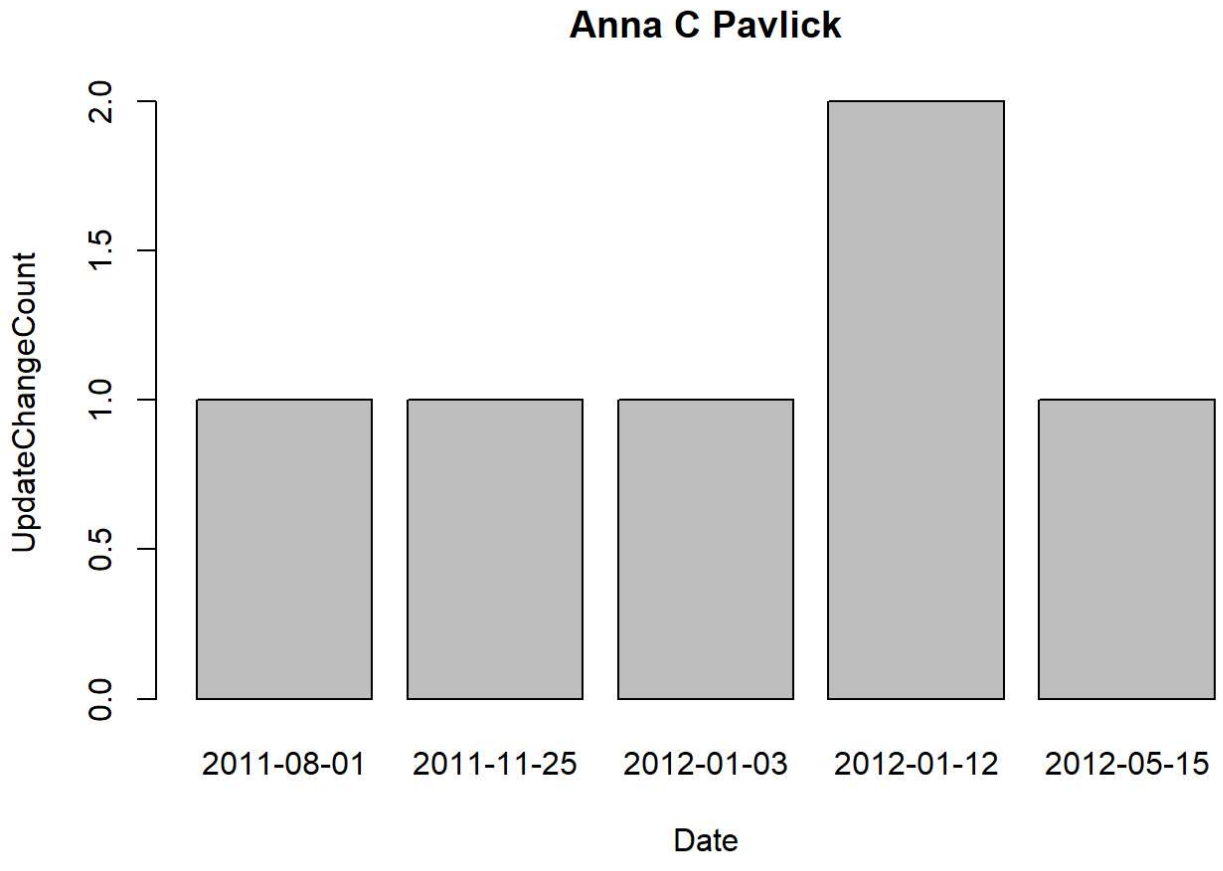

Adam Litterman
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

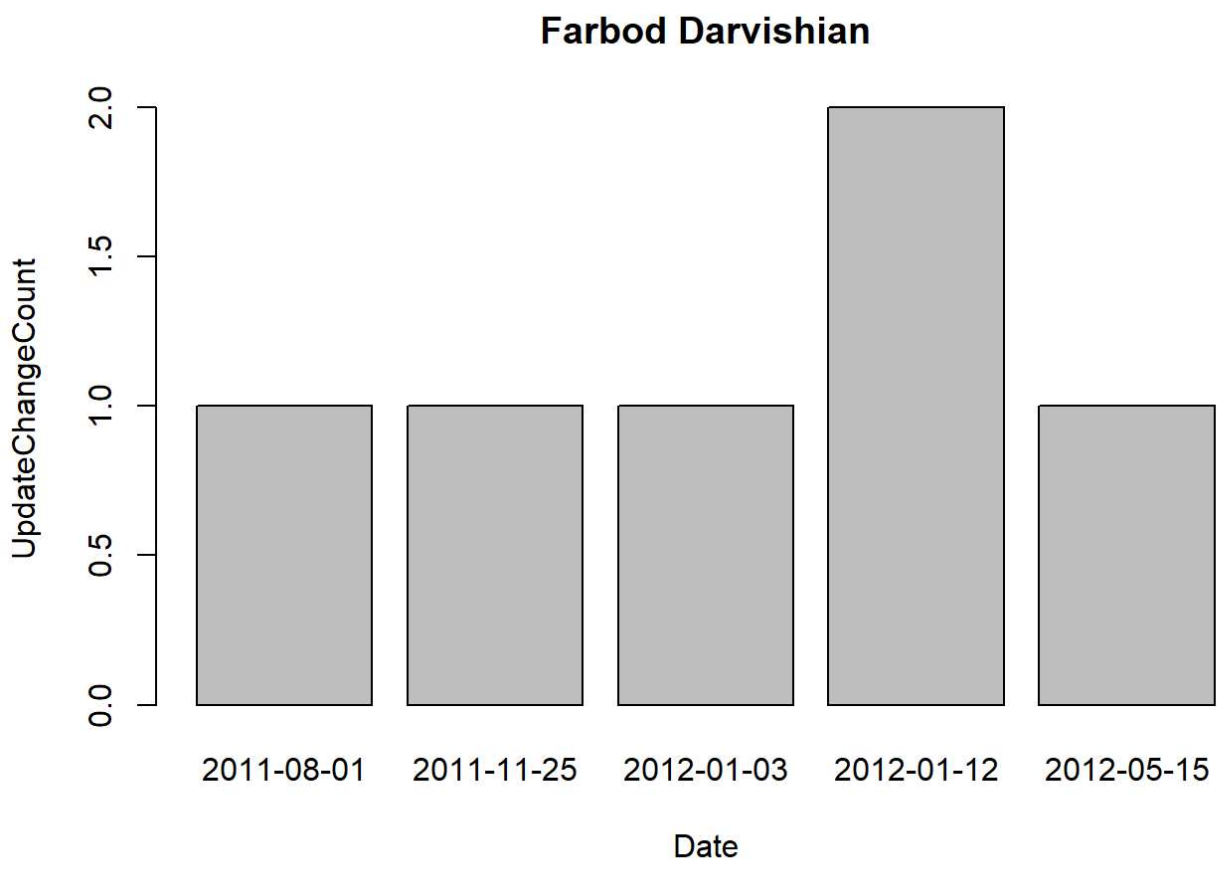

Joanne Yoon
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

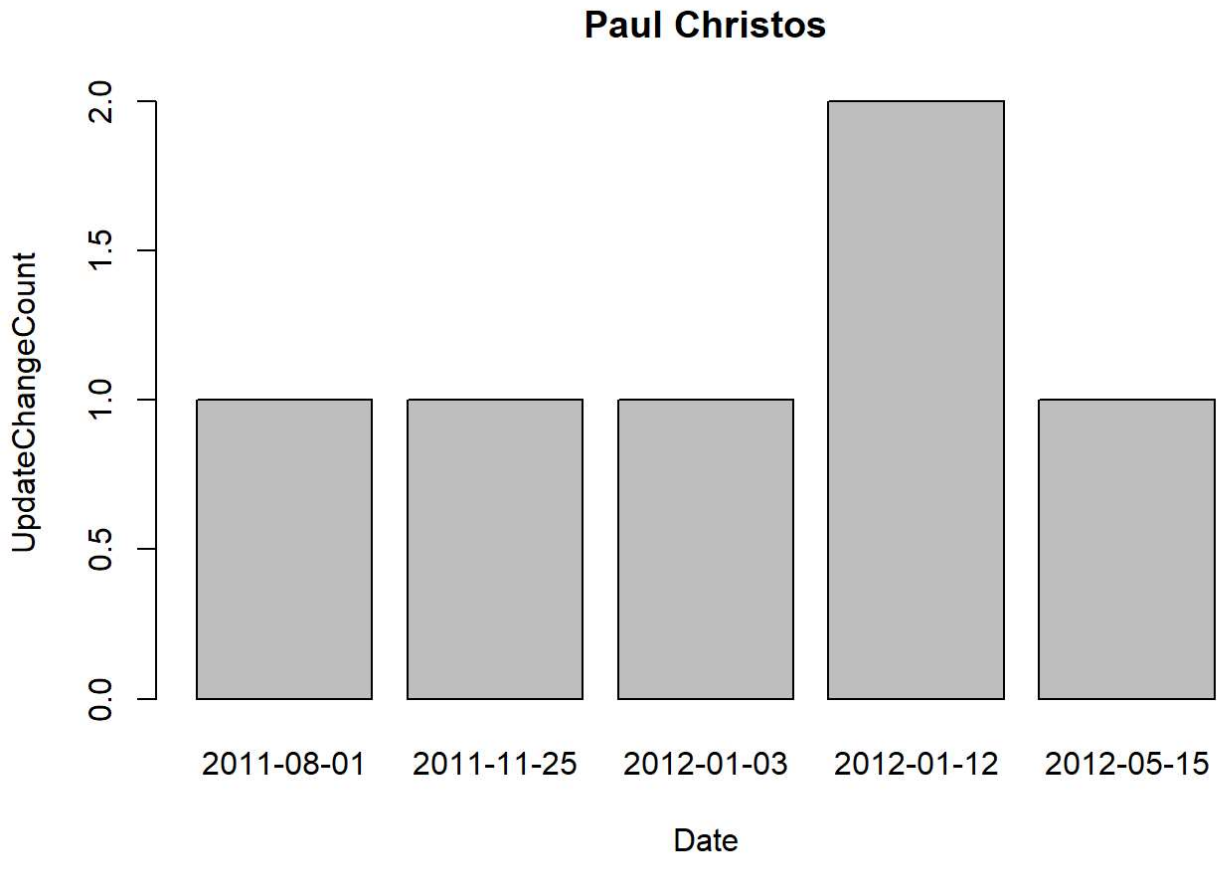

Elise Ng
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

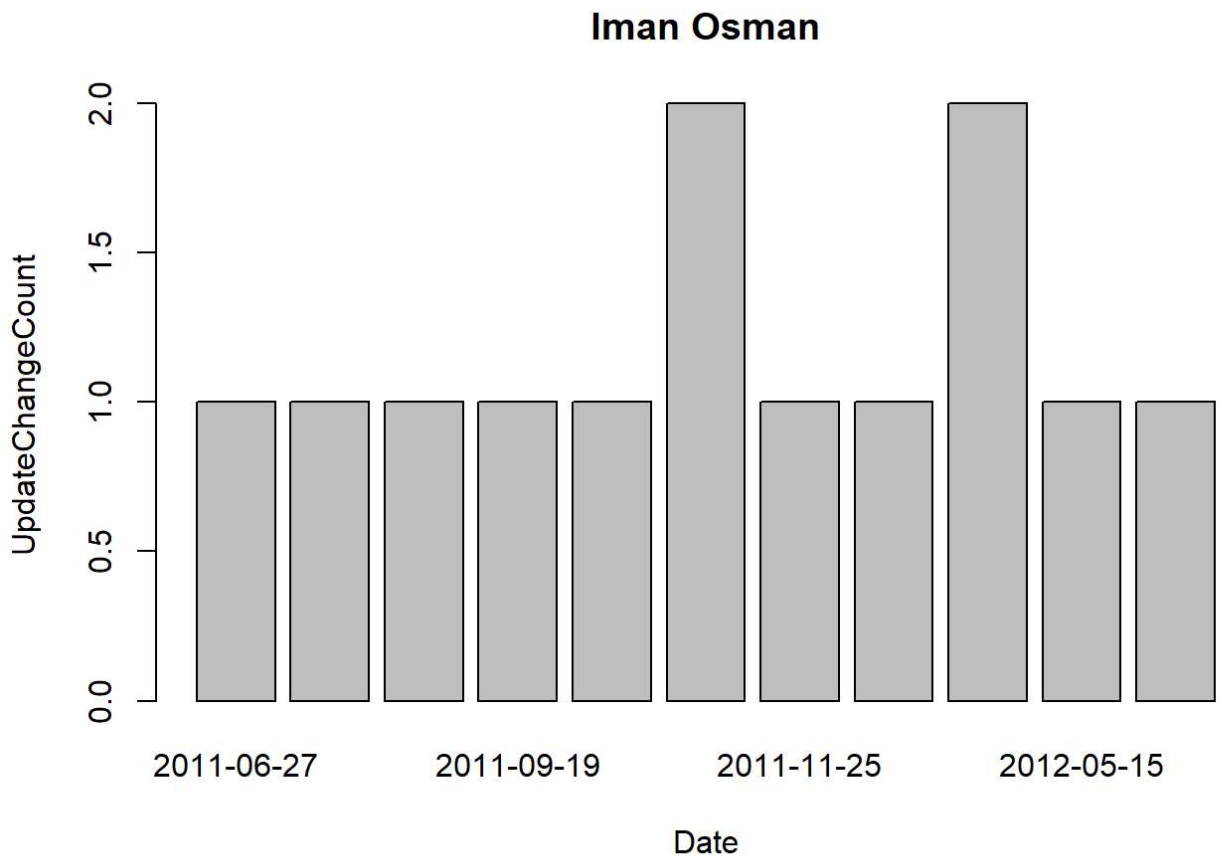

Richard L Shapiro
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

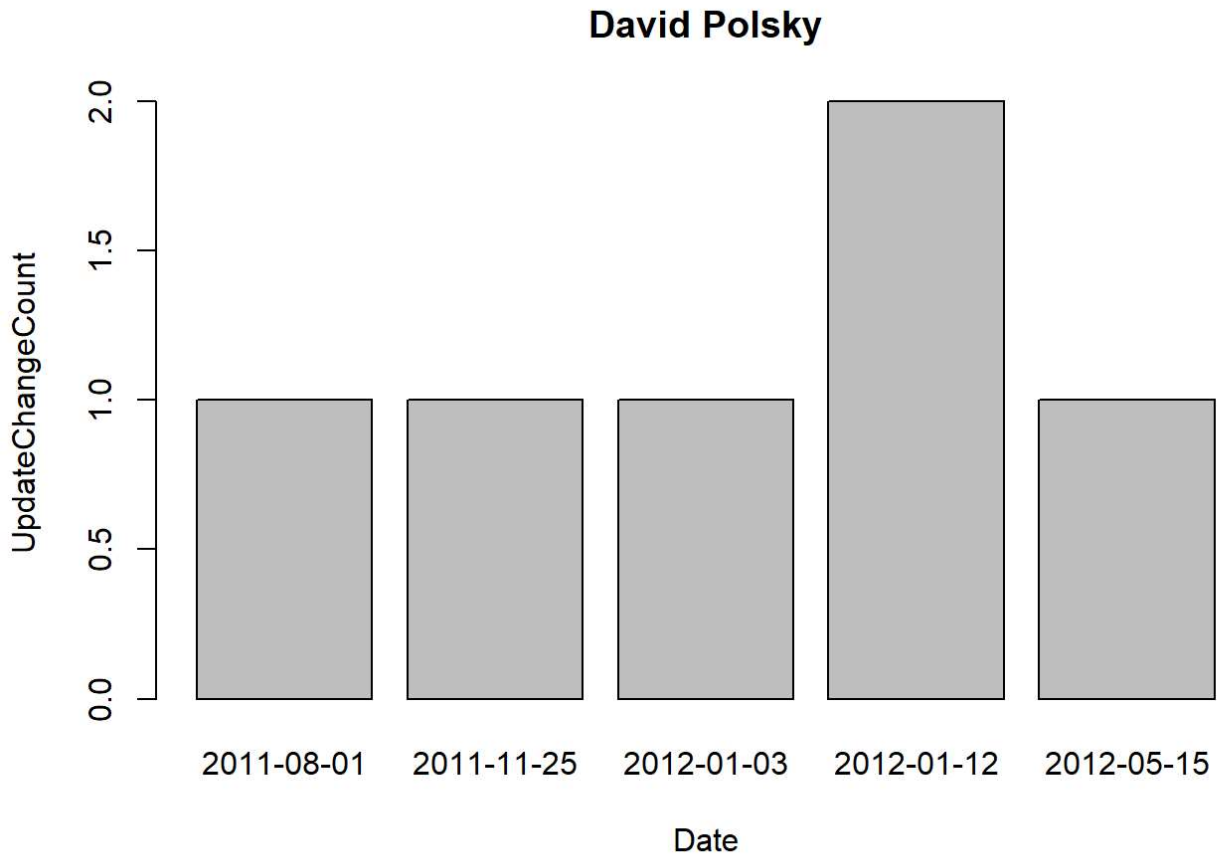

Russell S Berman
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

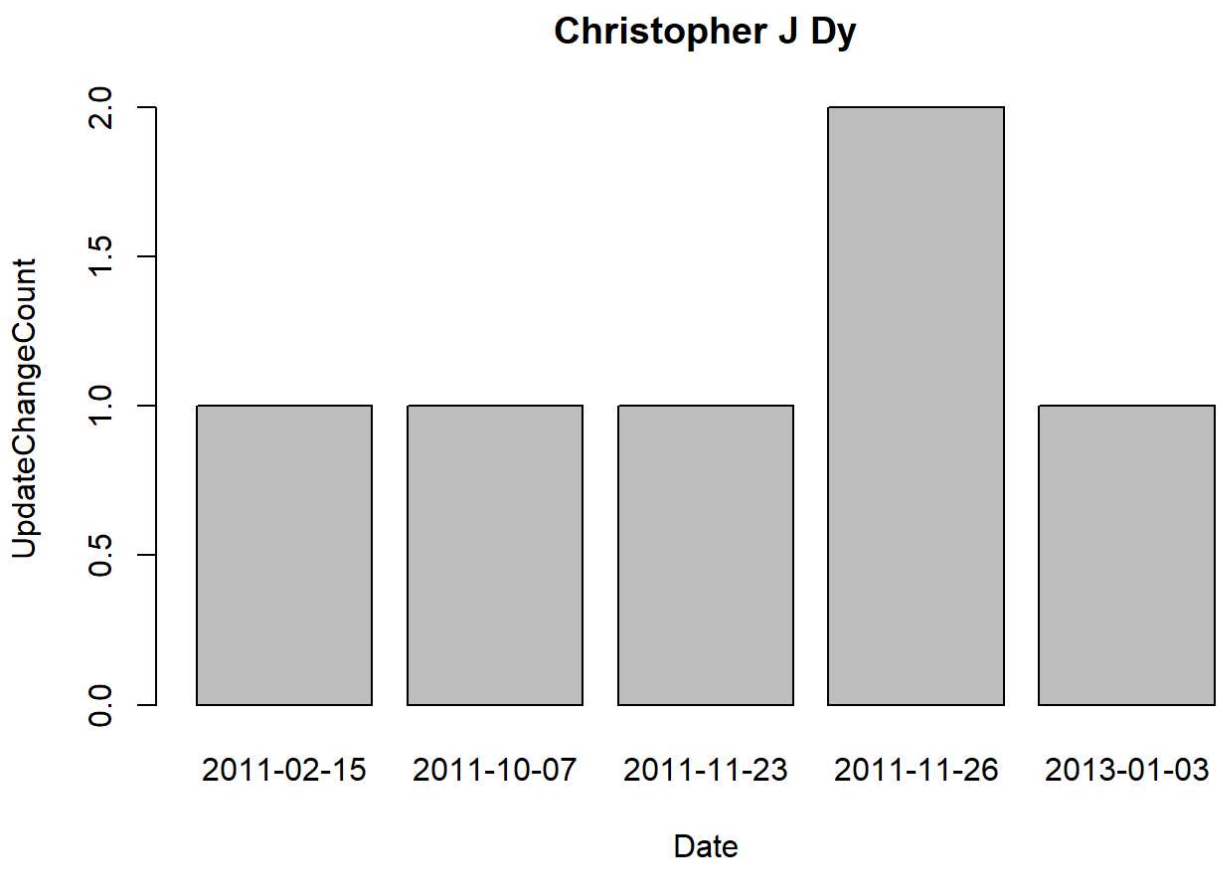

Anna C Pavlick
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

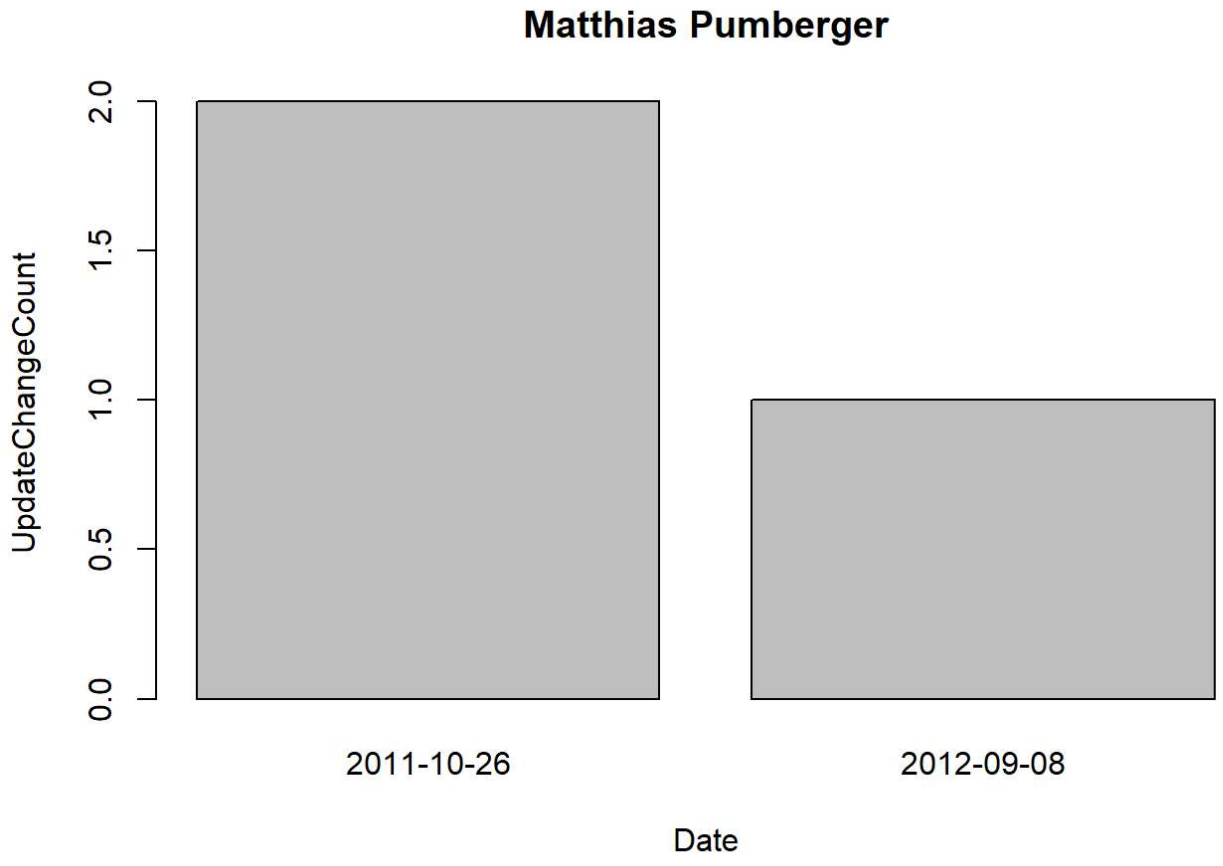

Farbod Darvishian
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

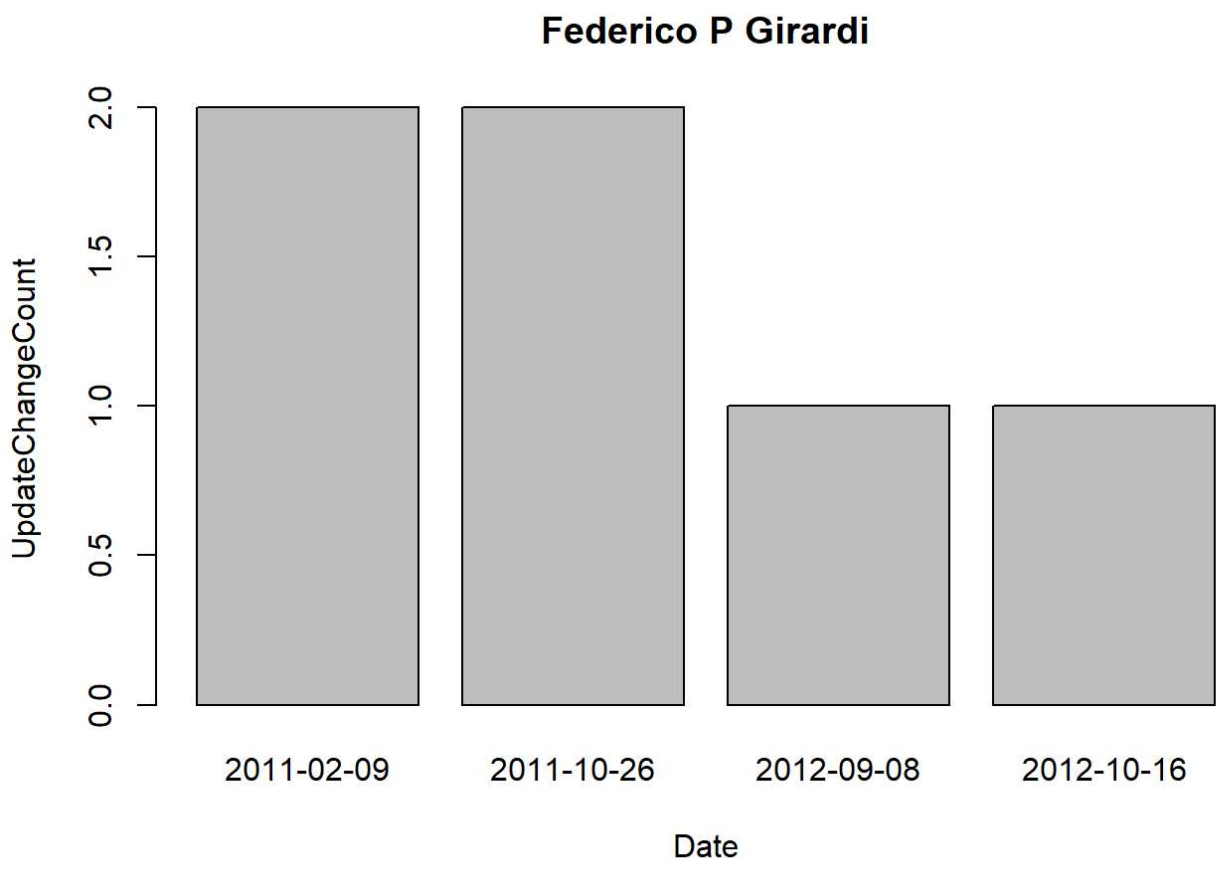

Paul Christos
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

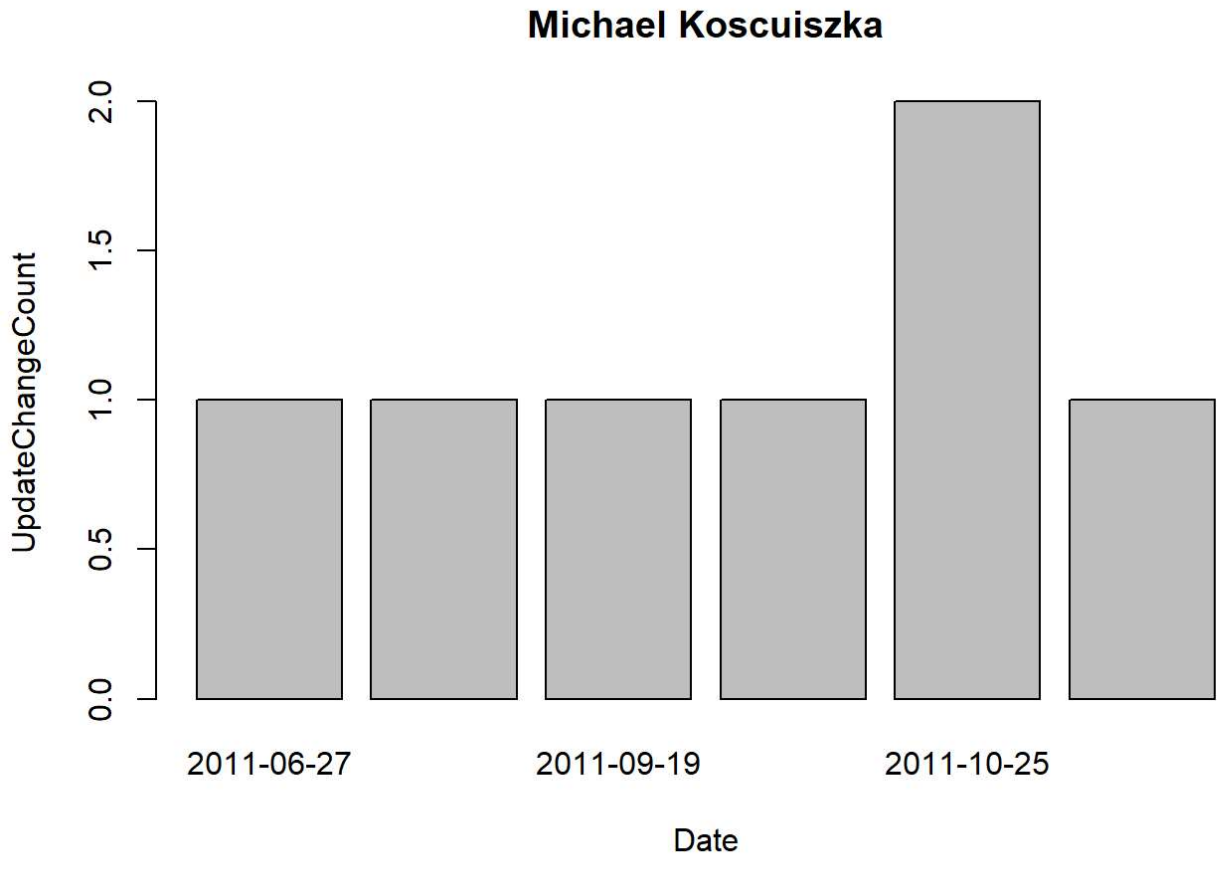

Iman Osman
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-11-25
2012-05-15
Date

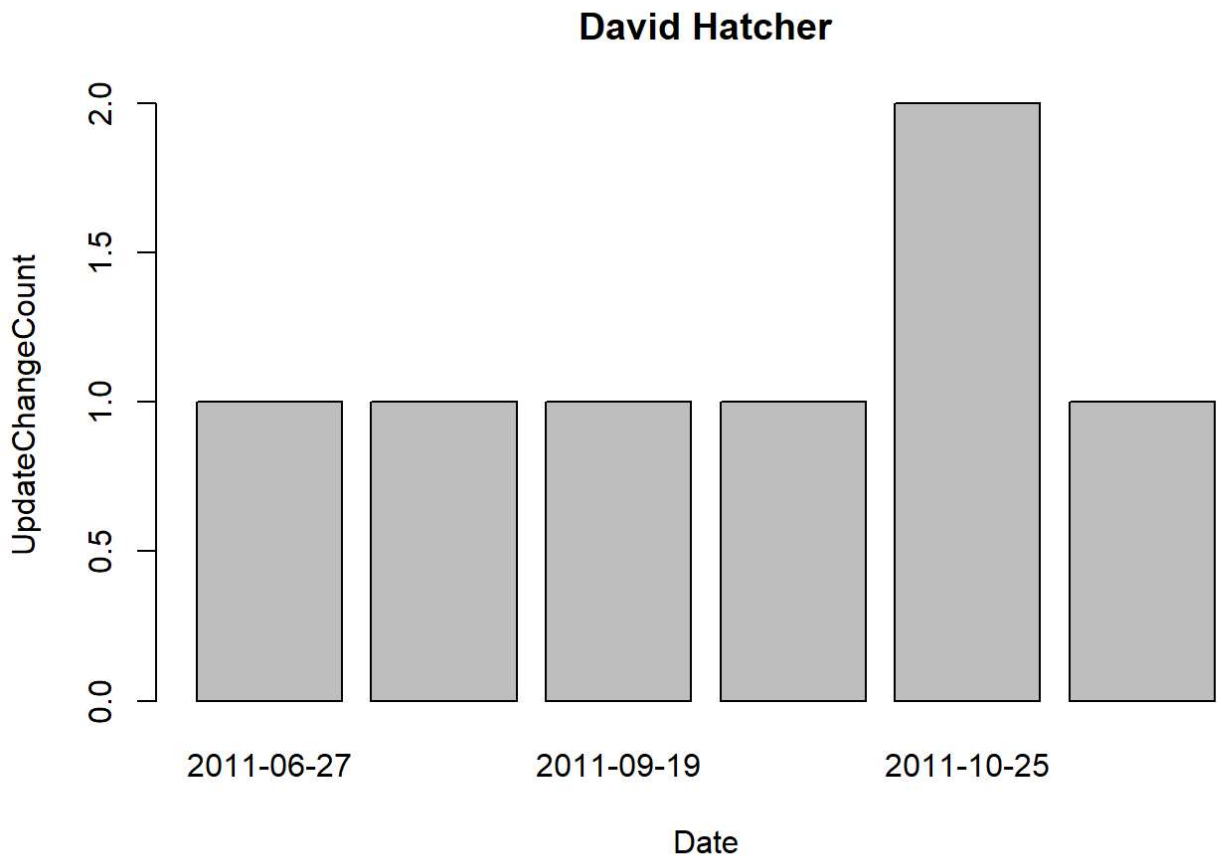

David Polsky
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-01
2011-11-25
2012-01-03
2012-01-12
2012-05-15
Date

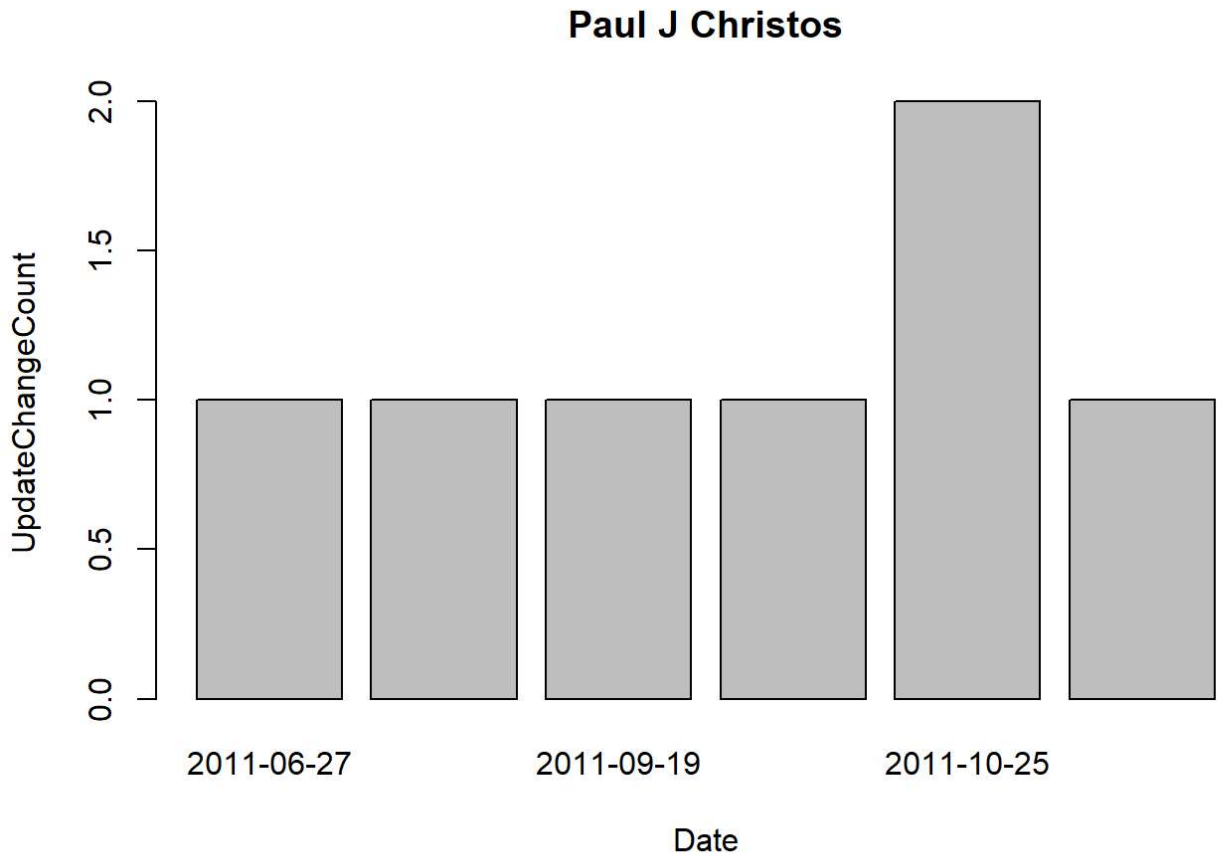

Christopher J Dy
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-02-15
2011-10-07
2011-11-23
2011-11-26
2013-01-03
Date

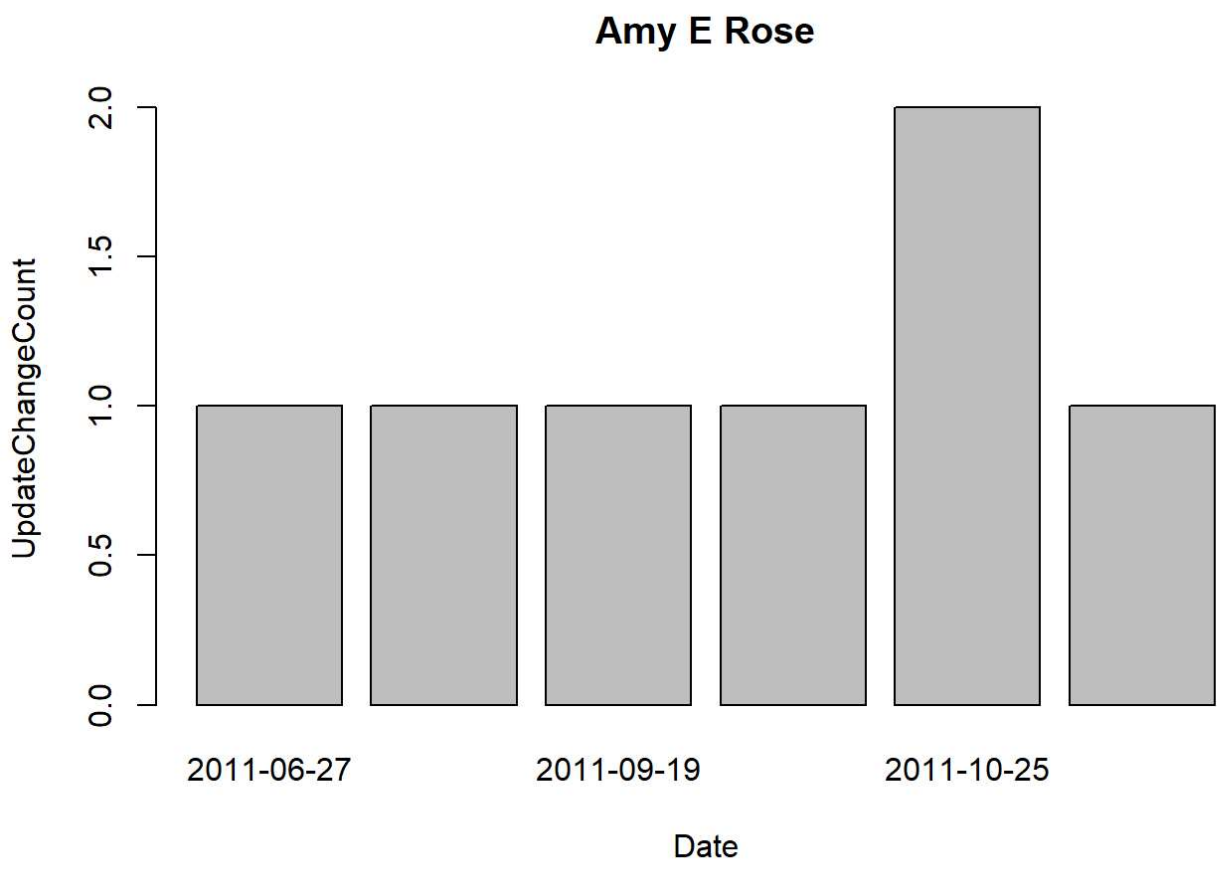

Matthias Pumberger
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-10-26
2012-09-08
Date

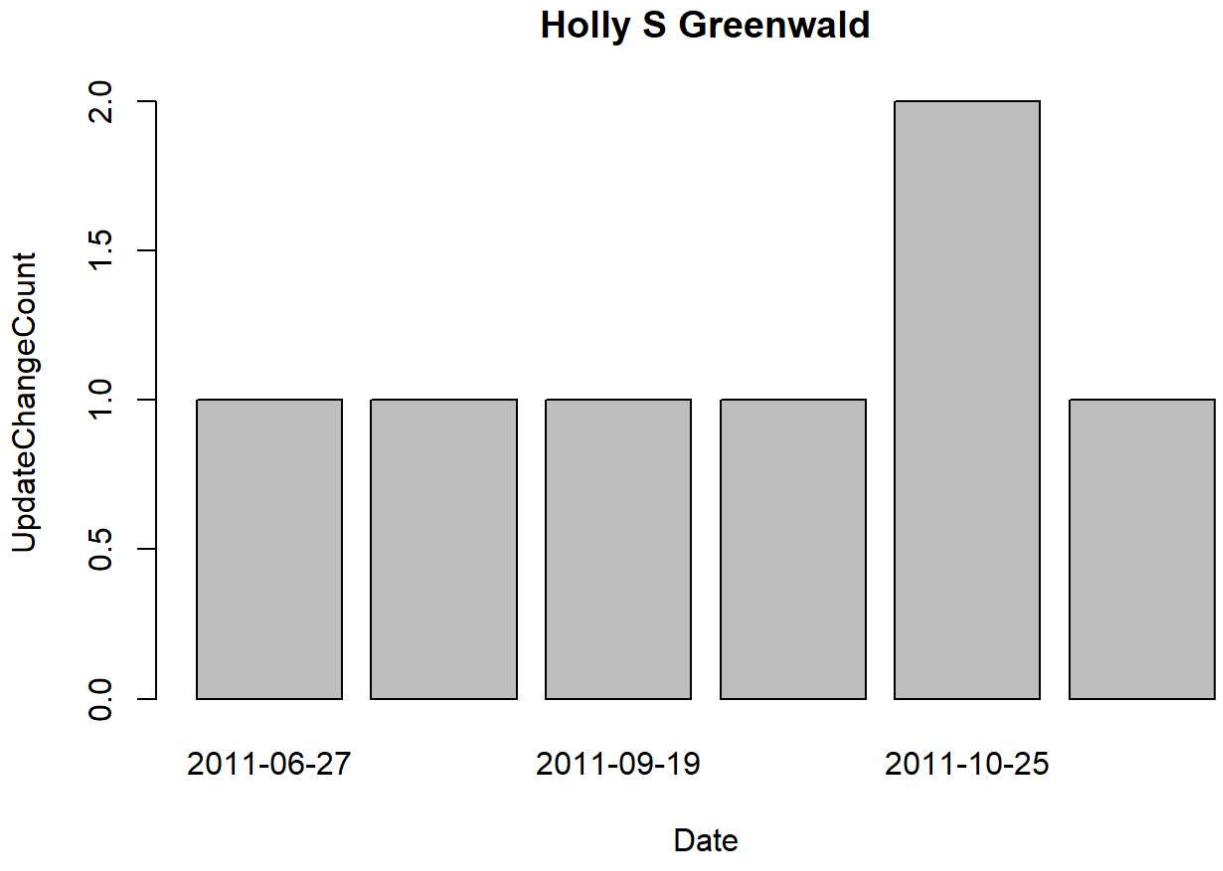

Federico P Girardi
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-02-09
2011-10-26
2012-09-08
2012-10-16
Date

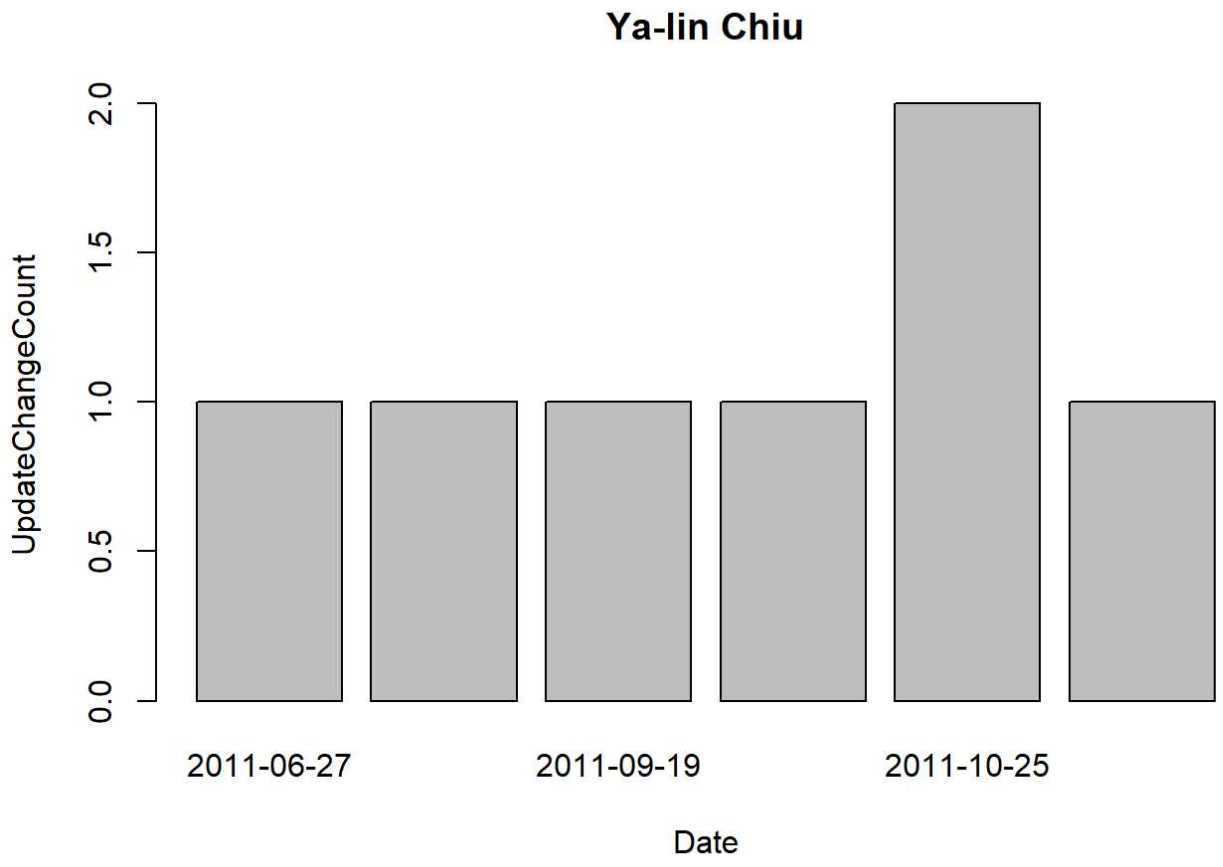

Michael Koscuiszka
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

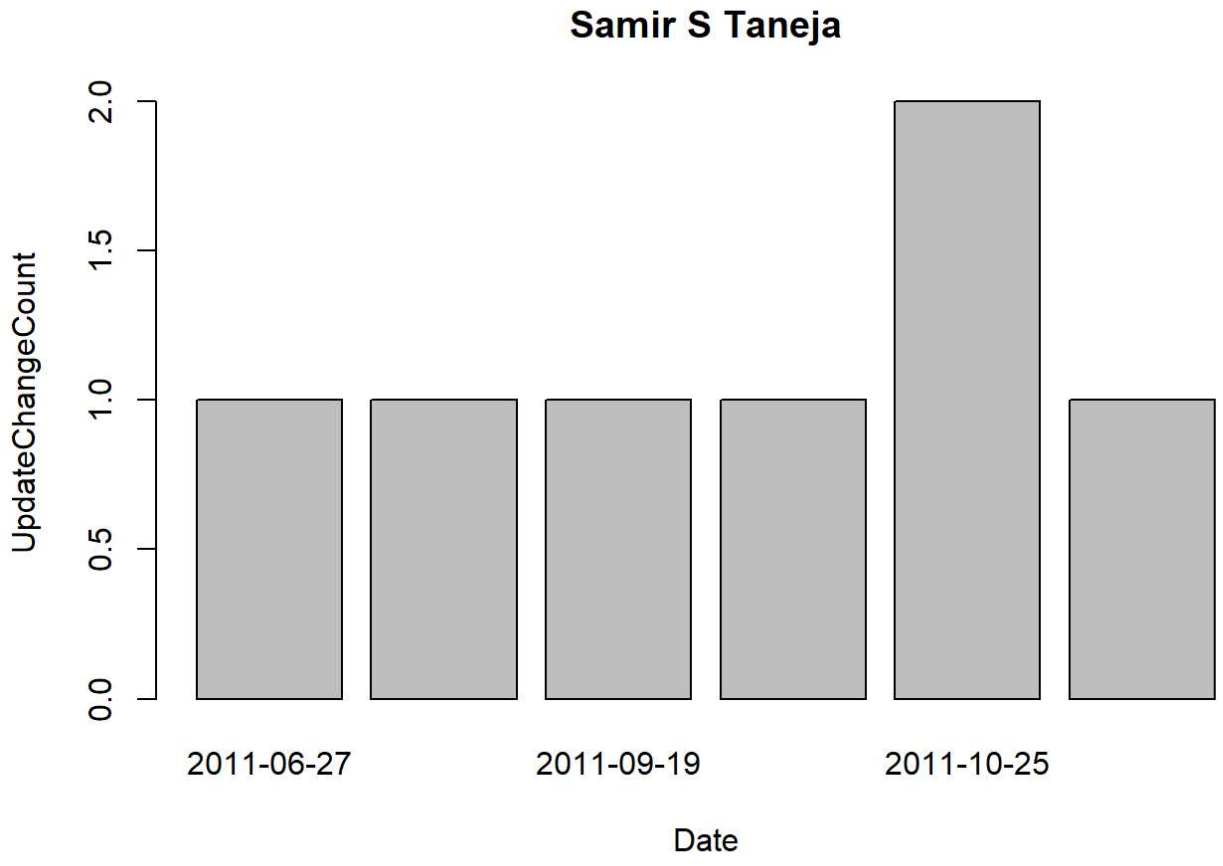

David Hatcher
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

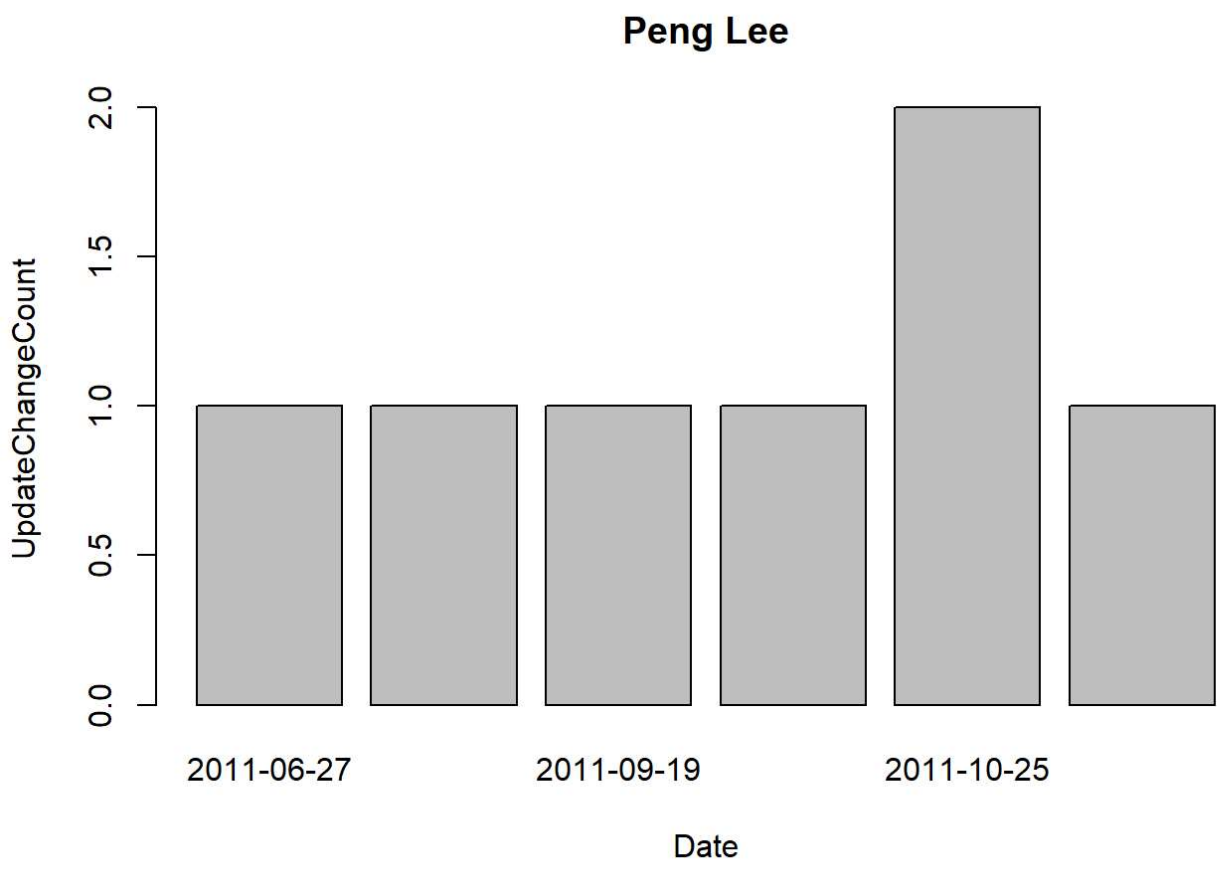

Paul J Christos
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

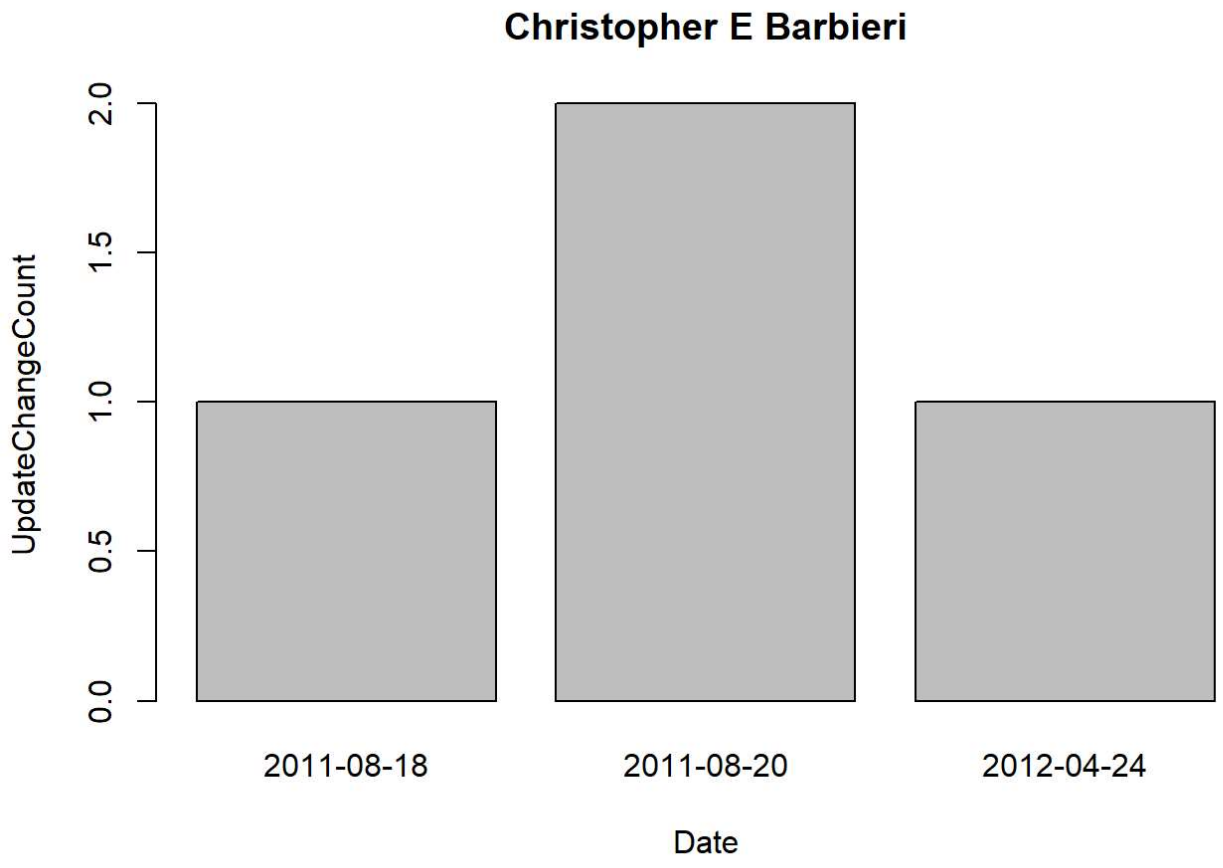

Amy E Rose
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

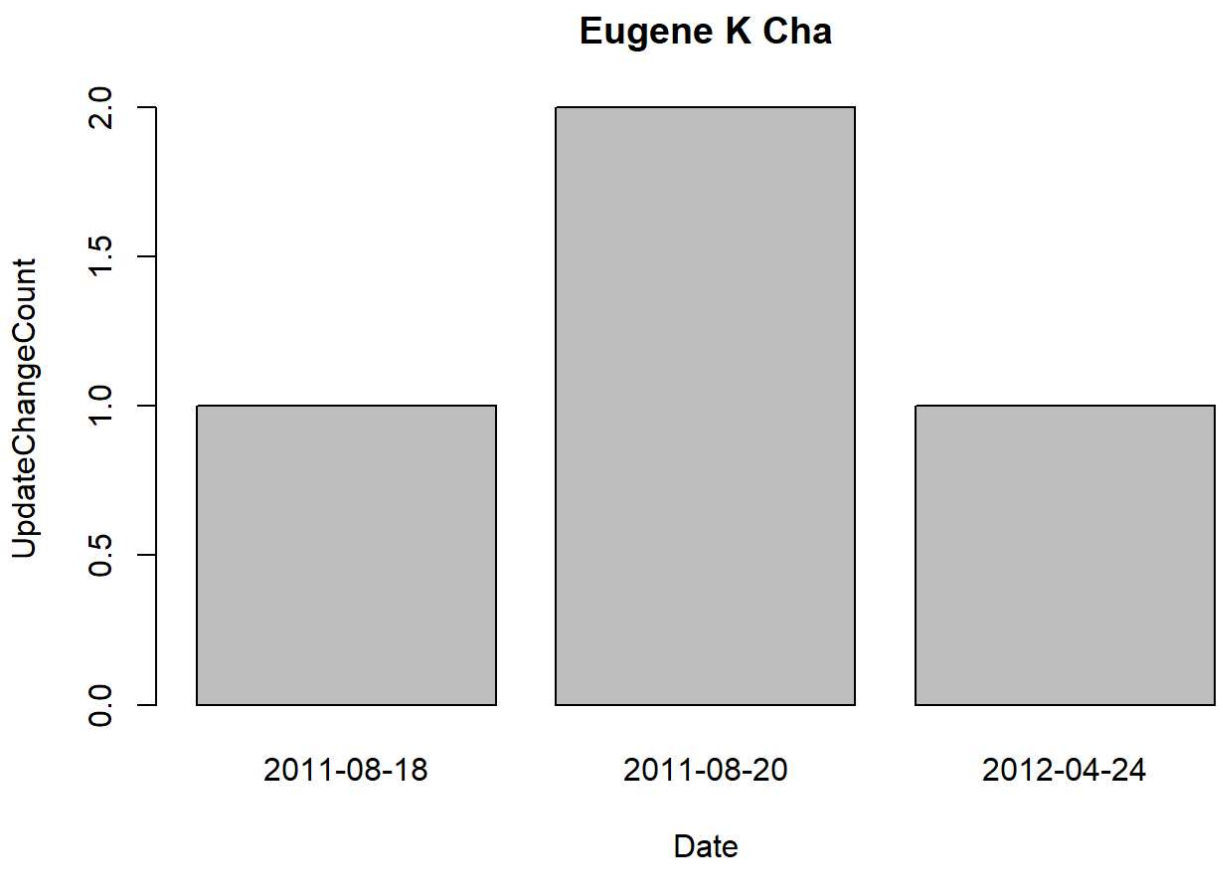

Holly S Greenwald
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

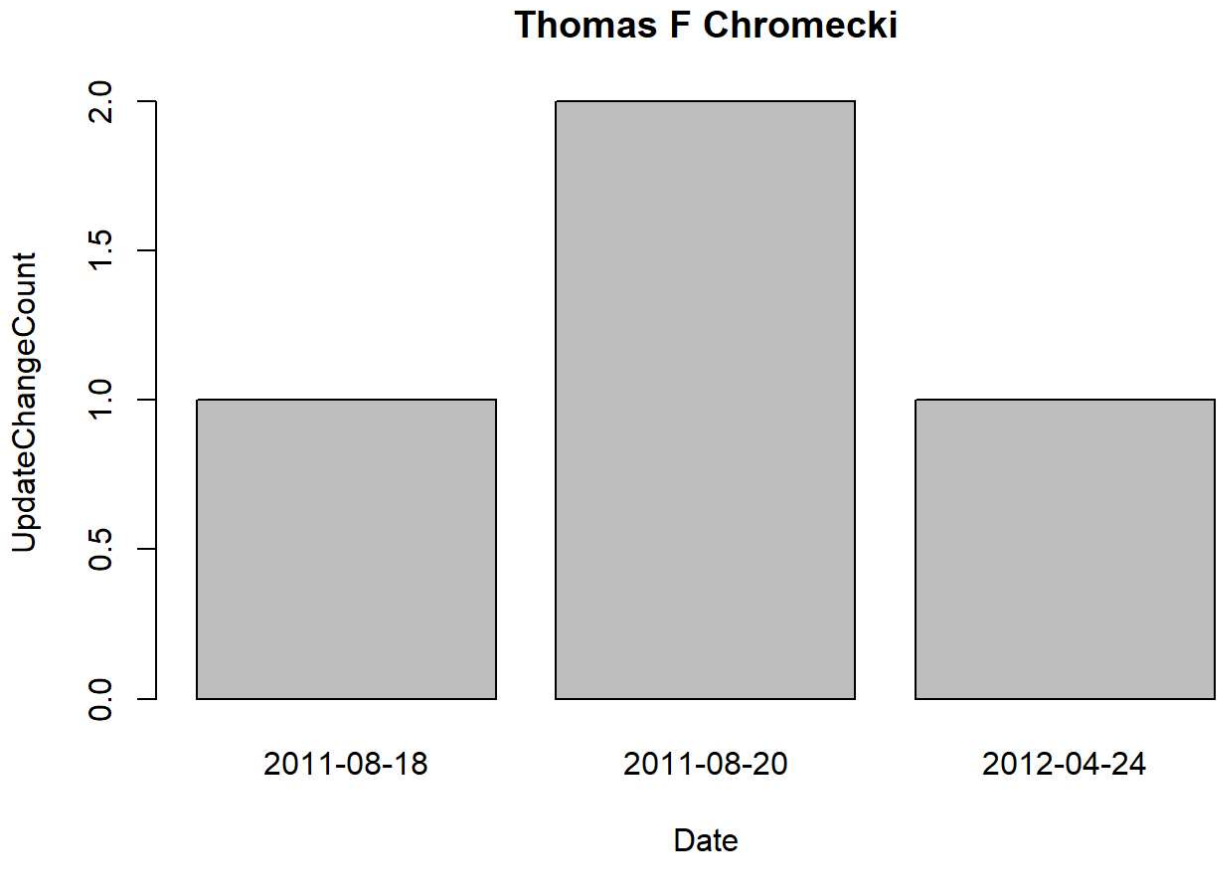

Ya-lin Chiu
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

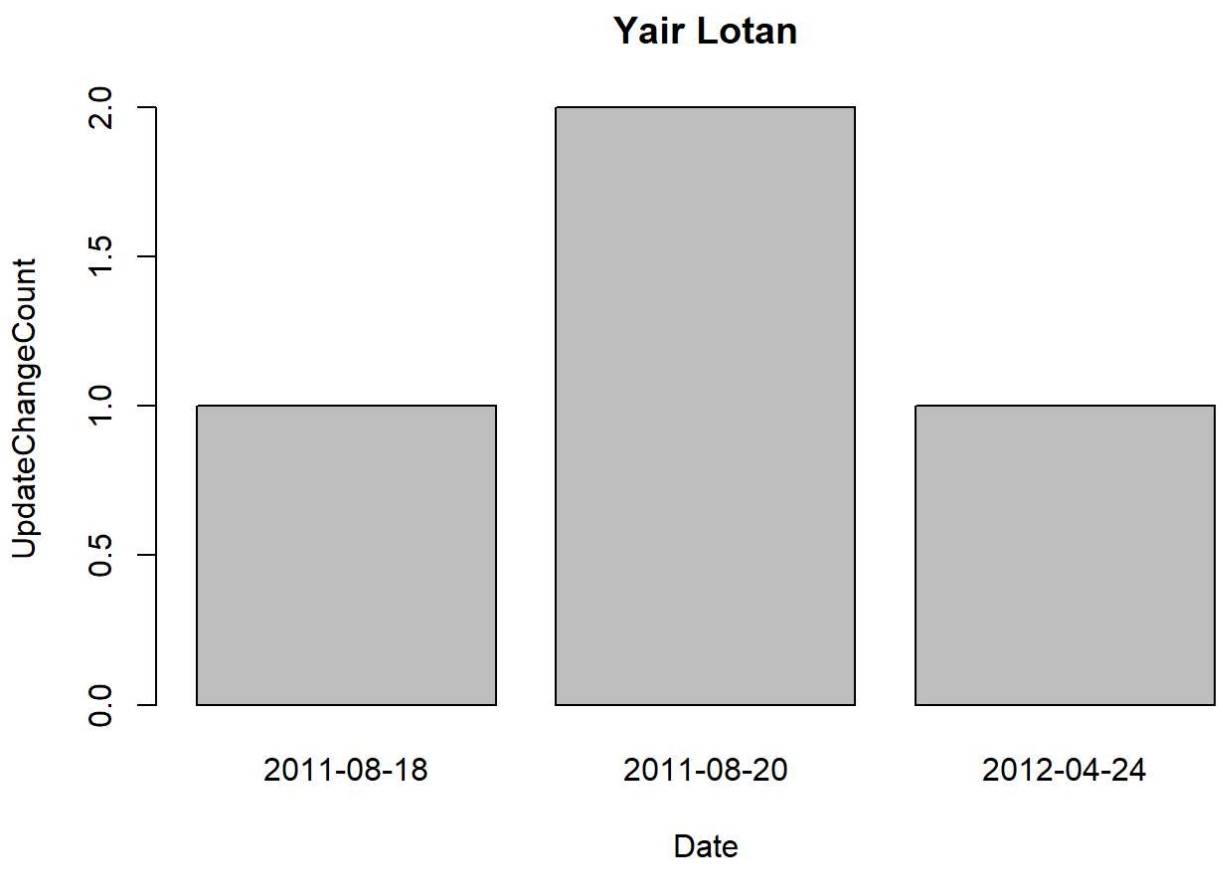

Samir S Taneja
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

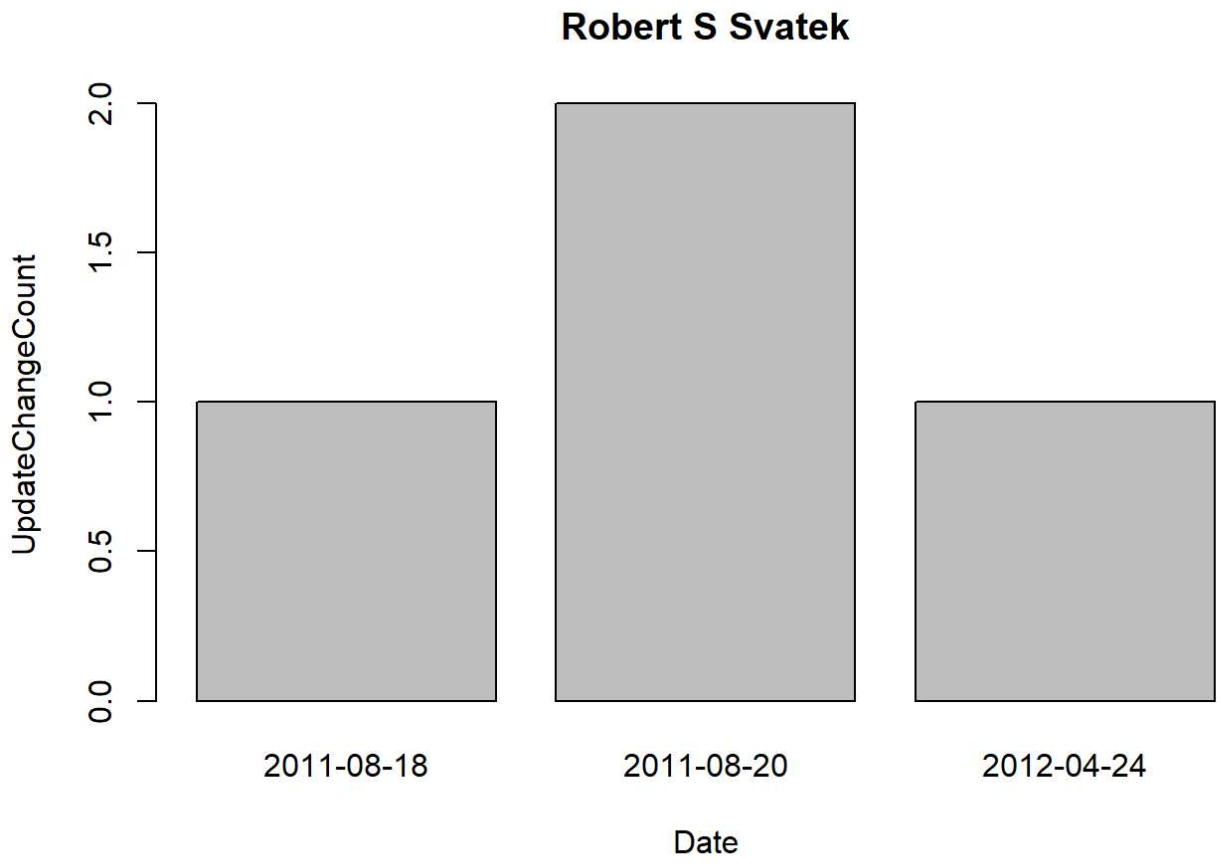

Peng Lee
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-06-27
2011-09-19
2011-10-25
Date

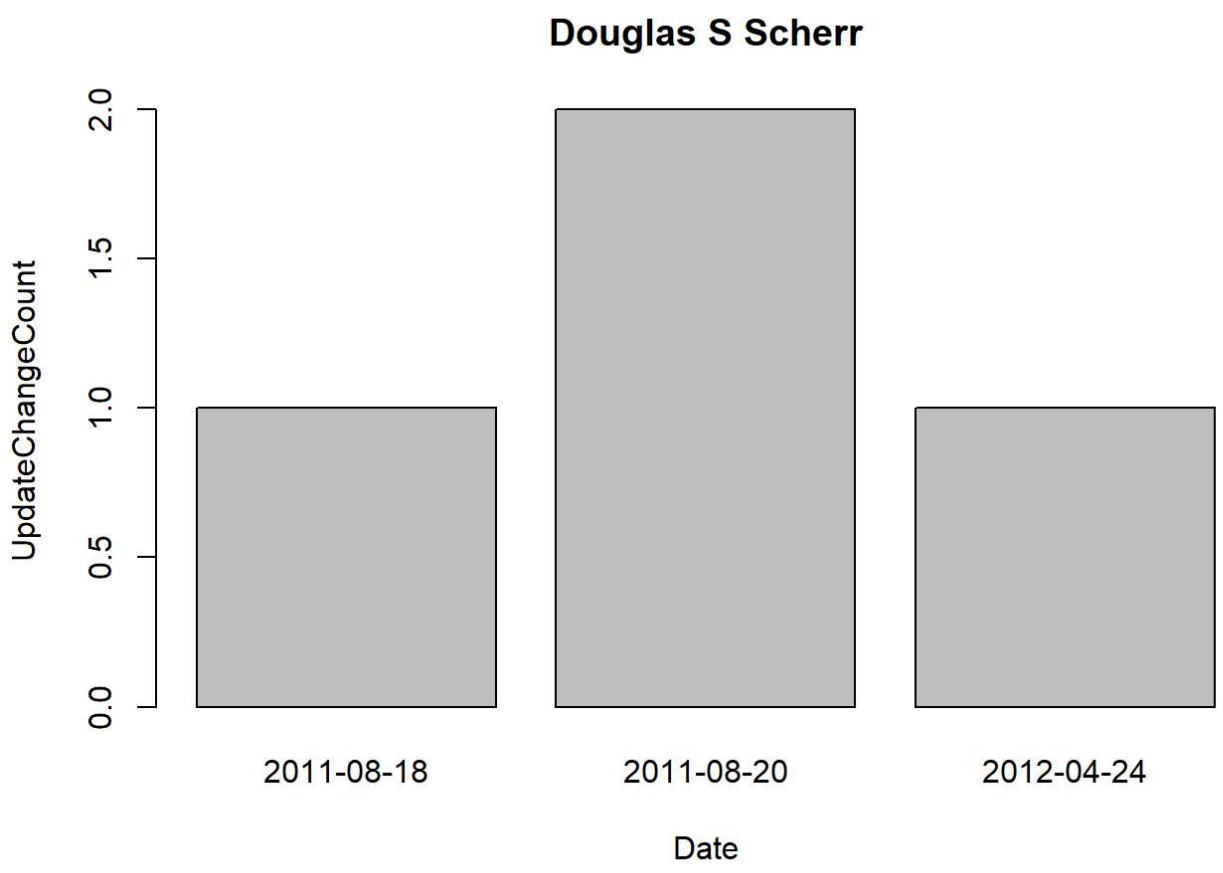

Christopher E Barbieri
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date

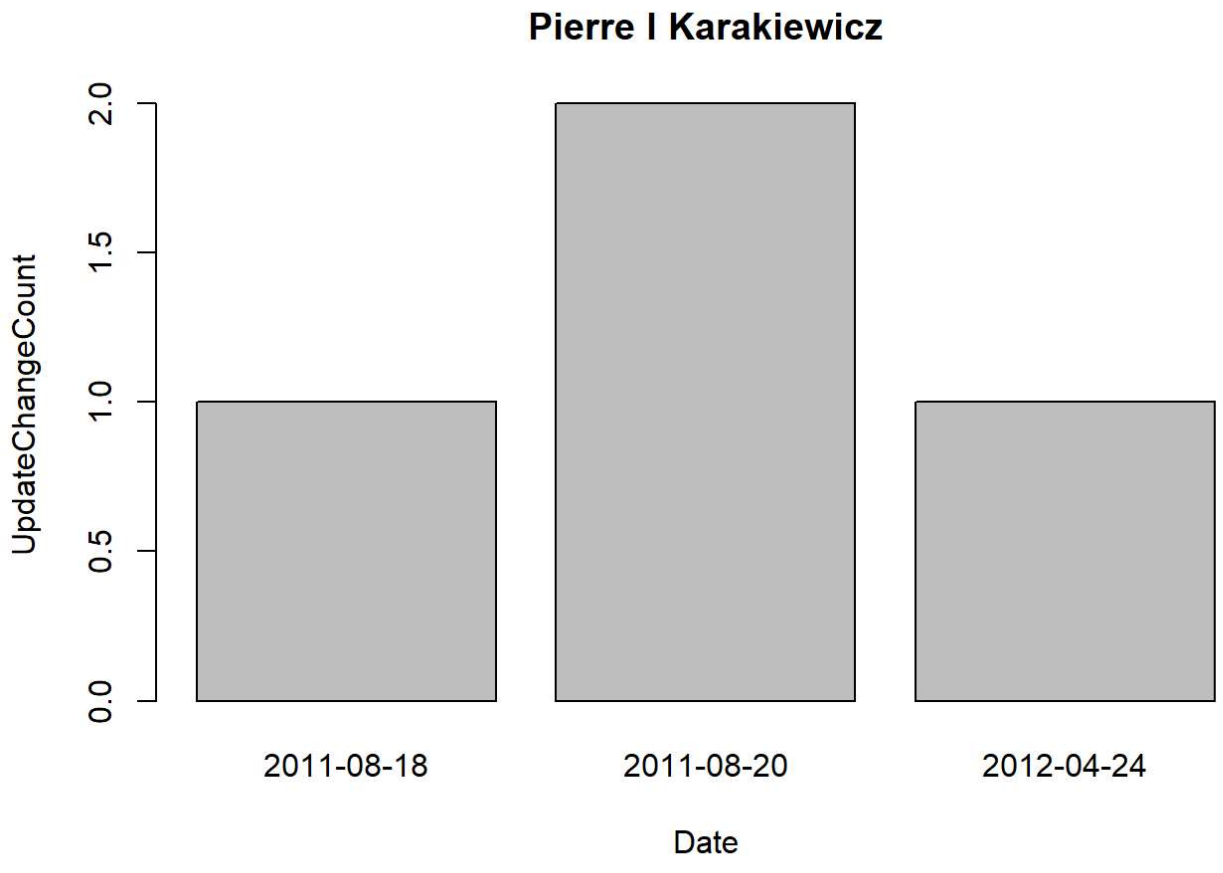

Eugene K Cha
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date

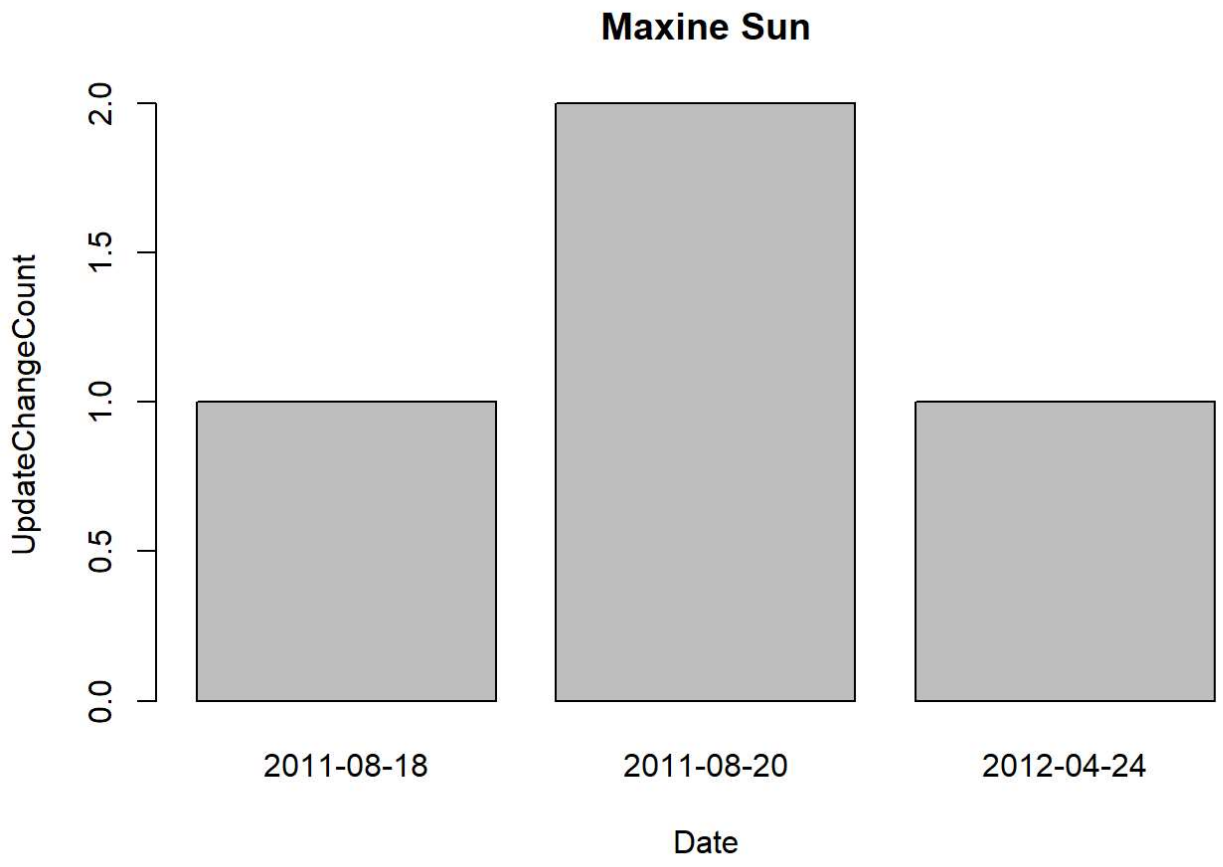

Thomas F Chromecki
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date

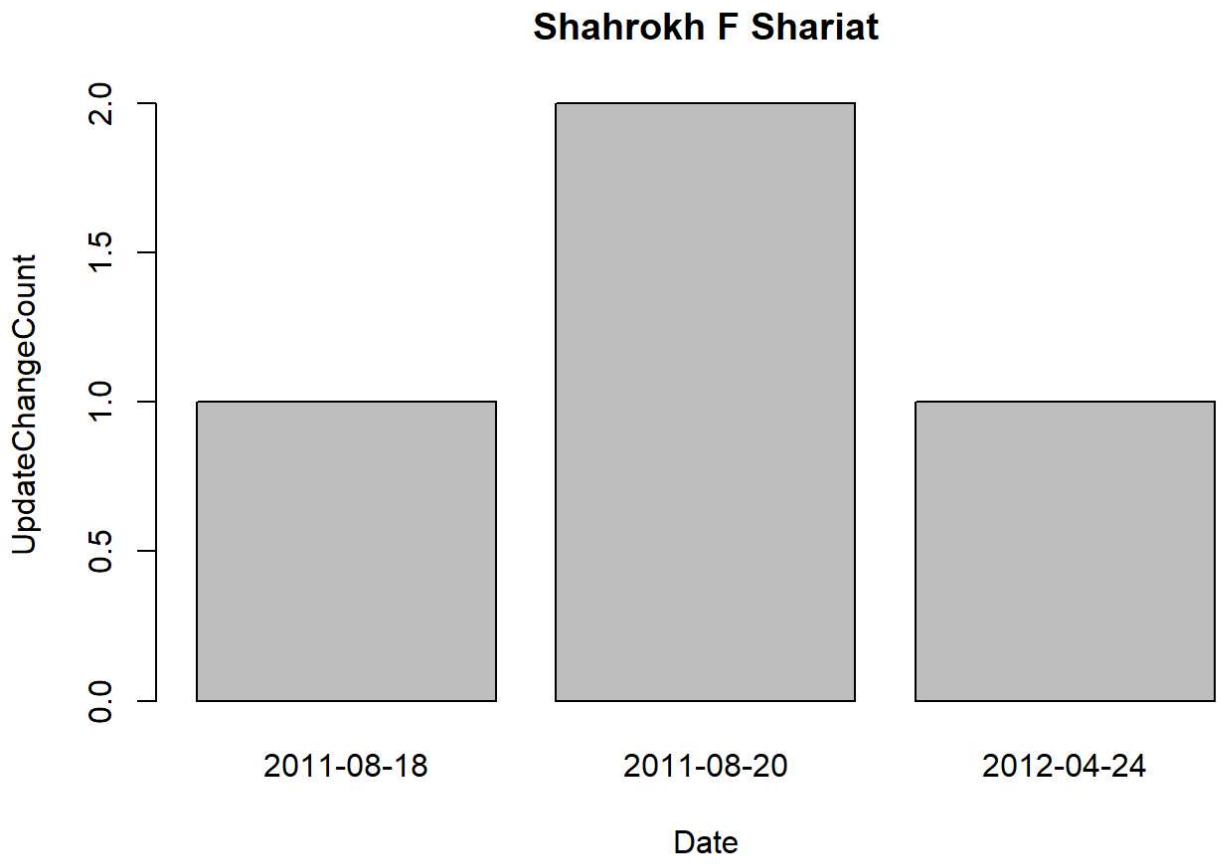

Yair Lotan
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date

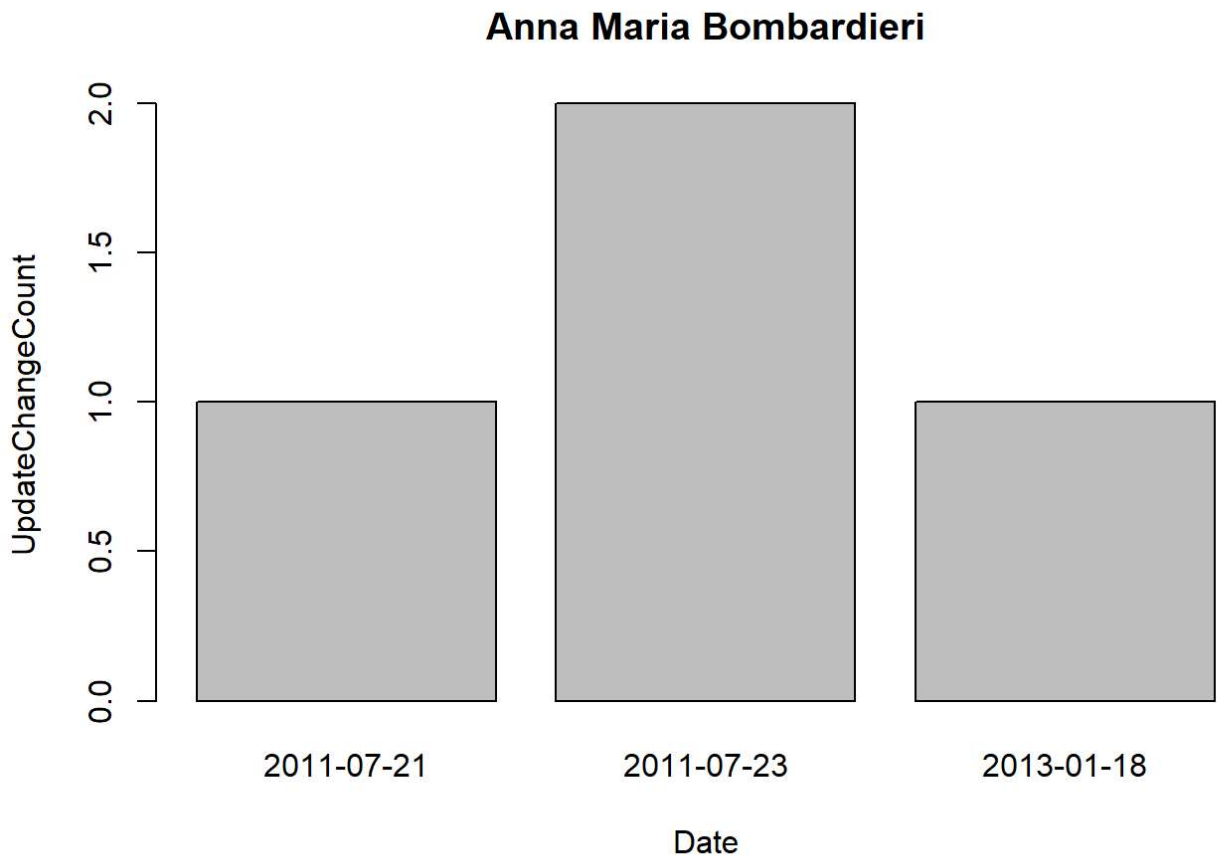

Robert S Svatek
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date

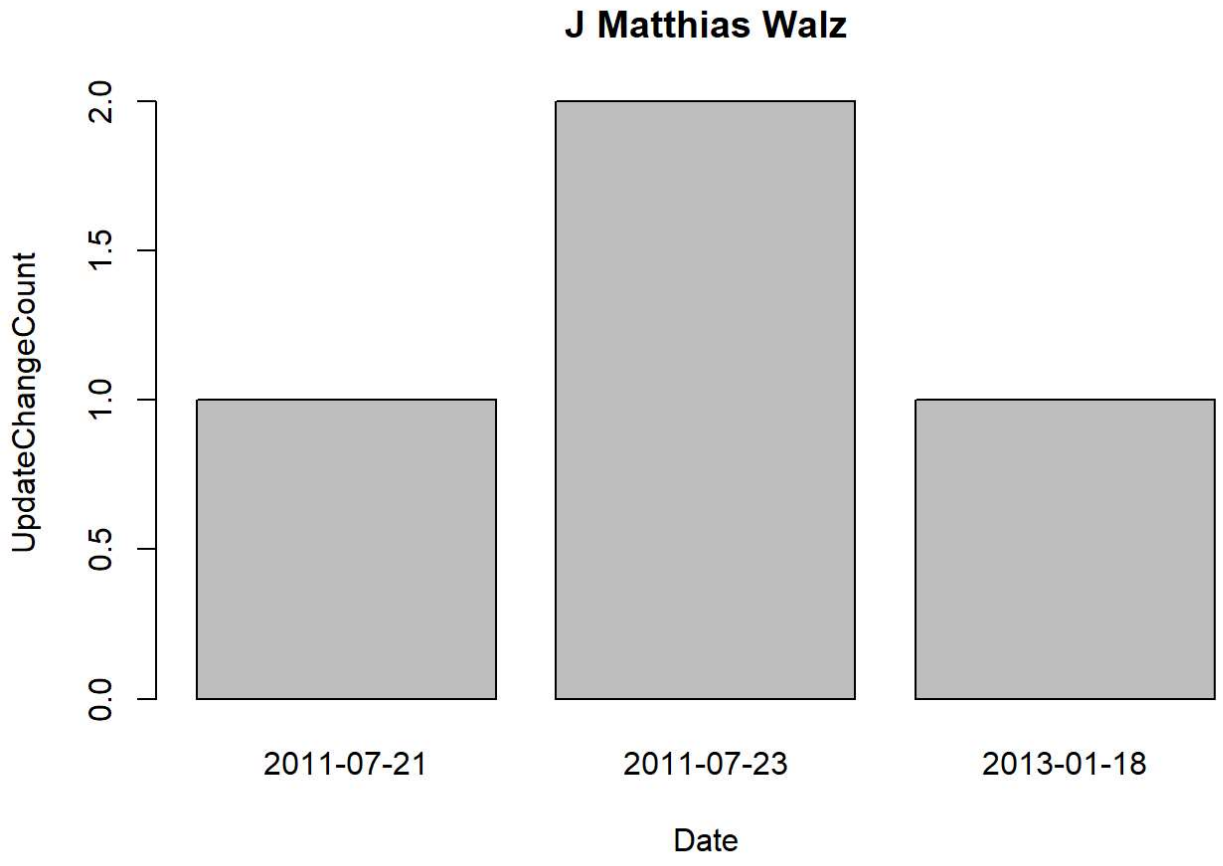

Douglas S Scherr
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date

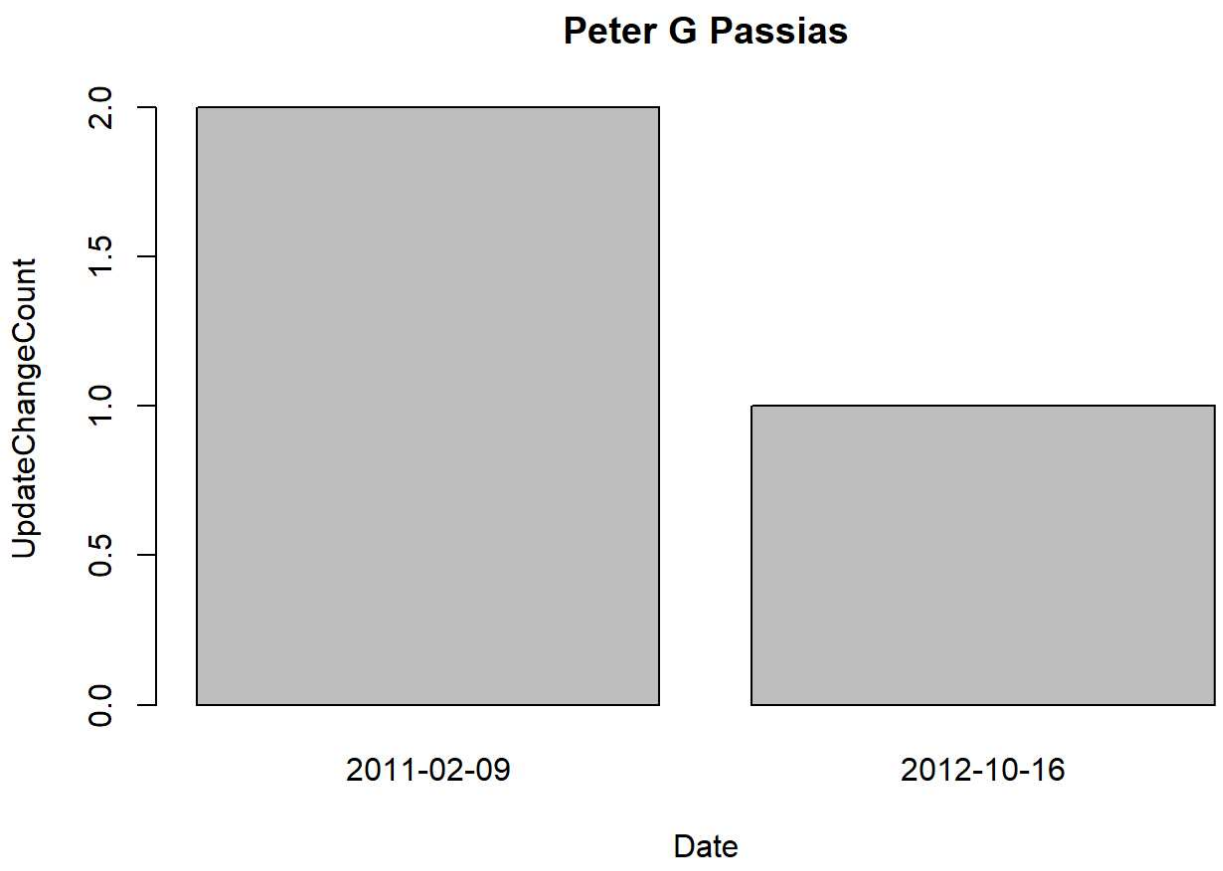

Pierre I Karakiewicz
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date


Maxine Sun
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date


Shahrokh F Shariat
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-08-18
2011-08-20
2012-04-24
Date


Anna Maria Bombardieri
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-07-21
2011-07-23
2013-01-18
Date


J Matthias Walz
UpdateChangeCount
0.0
0.5
1.0
1.5
2.0
2011-07-21
2011-07-23
2013-01-18
Date




```
select Quarter,sum(UpdateChangeCount) as Total from AuthorSummaryTable group by Quarter limit 20;
```

4 records

Quarter	Total
1	164
2	116
3	201
4	180

We found that during the 3rd Quarter the most Publication occurred and shown in the above Table. This Statement can also be backed up by the Plot of Different Author which show the seasonality pattern of their Publications