MWSUG 2016 - Paper TT02

Removing Duplicates Using SAS®

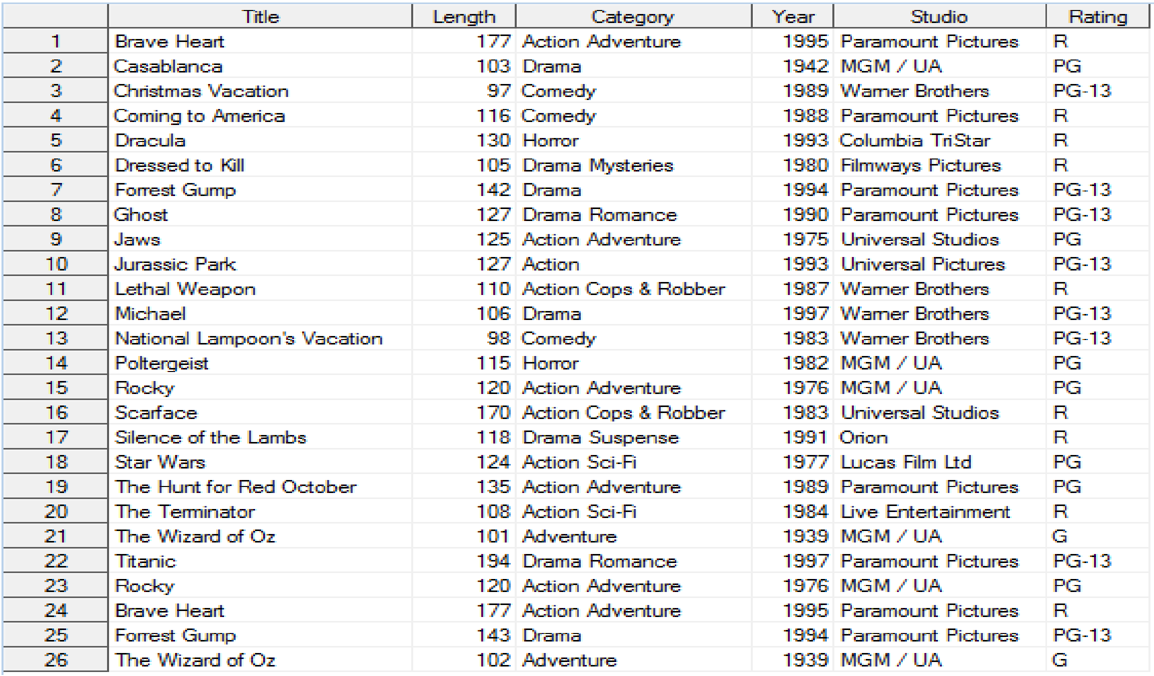
Kirk Paul Lafler, Software Intelligence Corporation, Spring Valley, California

Abstract

We live in a world of data – small data, big data, and data in every conceivable size between small and big. In today's world data finds its way into our lives wherever we are. We talk about data, create data, read data, transmit data, receive data, and save data constantly during any given hour in a day, and we still want and need more. So, we collect even more data at work, in meetings, at home, using our smartphones, in emails, in voice messages, sifting through financial reports, analyzing profits and losses, watching streaming videos, playing computer games, comparing sports teams and favorite players, and countless other ways. Data is growing and being collected at such astounding rates all in the hopes of being able to better understand the world around us. As SAS professionals, the world of data offers many new and exciting opportunities, but also presents a frightening realization that data sources may very well contain a host of integrity issues that need to be resolved first. This presentation describes the available methods to remove duplicate observations (or rows) from data sets (or tables) based on the row's values and/or keys using SAS®.

Introduction

An issue found in some data sets is the presence of duplicate observations and/or duplicate keys. When found, SAS can be used to remove any unwanted data. **Note:** Before duplicates are removed, be sure to consult with your organization's data analyst or subject matter expert to see if removal is necessary or permitted. It's better to be safe than sorry. This paper illustrates three very different approaches to remove duplicate observations (or rows) from data sets (or tables) based on the observation's values and/or keys using SAS®. Each example is illustrated using a single data set, MOVIES. The Movies data set contains 26 observations, and has a structure consisting of six columns. Title, Category, Studio, and Rating are defined as character columns; and Length and Year are defined as numeric columns. The Movies data set contains two duplicate observations – Brave Heart and Rocky; and two duplicate Title keys – Forrest Gump and The Wizard of Oz, shown below.



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
9	Jaws	125	Action Adventure	1975	Universal Studios	PG
10	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
11	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
12	Michael	106	Drama	1997	Warner Brothers	PG-13
13	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
14	Poltergeist	115	Horror	1982	MGM / UA	PG
15	Rocky	120	Action Adventure	1976	MGM / UA	PG
16	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
17	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
18	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
19	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
20	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
21	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
22	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13
23	Rocky	120	Action Adventure	1976	MGM / UA	PG
24	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
25	Forrest Gump	143	Drama	1994	Paramount Pictures	PG-13
26	The Wizard of Oz	102	Adventure	1939	MGM / UA	G

Method #1 – Using PROC SORT to Remove Duplicates

The first method, and one that is popular with SAS professionals everywhere, uses PROC SORT to remove duplicates. The SORT procedure supports three options for the removal of duplicates: **DUPOUT=**, **NODUPRECS**, and **NODUPKEYS**.

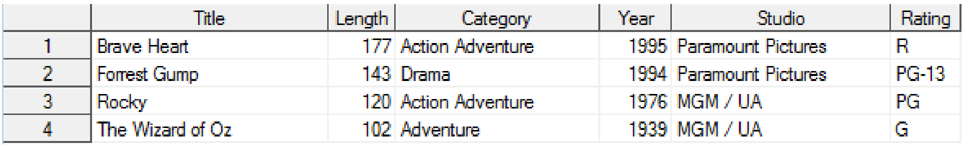
Specifying the DUPOUT= Option

PROC SORT's **DUPOUT=** option can be used to identify duplicate observations before actually removing them from a data set. The DUPOUT= option is used with either the NODUPKEYS or NODUPRECS option to name a data set that will contain duplicate keys or duplicate observations. The DUPOUT= option is generally used when the data set is too large for visual inspection. In the next code example, the DUPOUT= and NODUPKEY options are specified. The resulting output data set contains the duplicate observations for Brave Heart, Forrest Gump, Rocky and The Wizard of Oz.

PROC SORT Code

```
PROC SORT DATA=Movies
        DUPOUT=Movies_Sorted_Dupout_NoDupkey
        NODUPKEY ;
  BY Title ;
RUN ;
```

Resulting Table



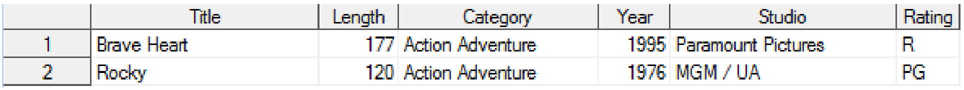
	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Forrest Gump	143	Drama	1994	Paramount Pictures	PG-13
3	Rocky	120	Action Adventure	1976	MGM / UA	PG
4	The Wizard of Oz	102	Adventure	1939	MGM / UA	G

In the next example, the **DUPOUT=** and **NODUPRECS** options are specified. The resulting output data set contains the duplicate observations for Brave Heart and Rocky because these rows have identical data for all columns.

PROC SORT Code

```
PROC SORT DATA=Movies
        DUPOUT=Movies_Sorted_Dupout_NoDupRecs
        NODUPRECS ;
  BY Title ;
RUN ;
```

Resulting Table



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Rocky	120	Action Adventure	1976	MGM / UA	PG

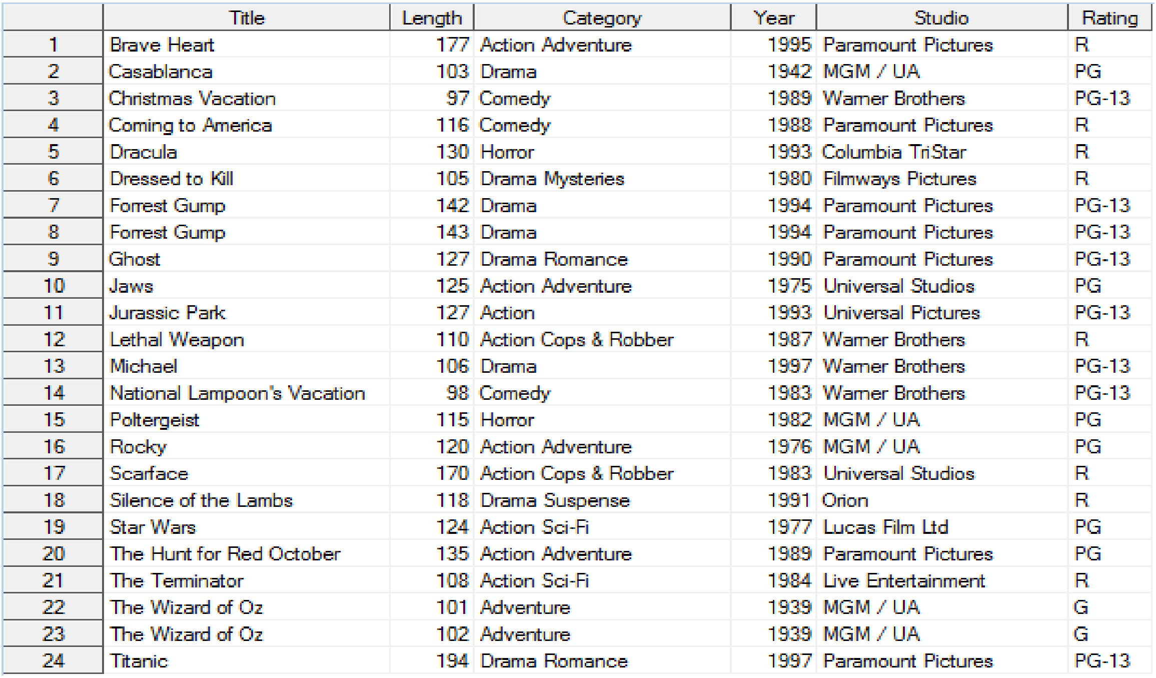
Specifying the NODUPRECS (or NODUP) Option

PROC SORT's **NODUPRECS** (or **NODUPREC**) (or **NODUP**) option identifies observations with identical values for all columns are removed from the output data set. The resulting output data saw the removal of the duplicate observations for Brave Heart and Rocky because they have identical data for all columns.

PROC SORT Code

```
PROC SORT DATA=Movies
           OUT=Movies_Sorted_without_DupRecs
          NODUPRECS ;
  BY Title ;
RUN ;
```

Resulting Table



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Forrest Gump	143	Drama	1994	Paramount Pictures	PG-13
9	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
10	Jaws	125	Action Adventure	1975	Universal Studios	PG
11	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
12	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
13	Michael	106	Drama	1997	Warner Brothers	PG-13
14	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
15	Poltergeist	115	Horror	1982	MGM / UA	PG
16	Rocky	120	Action Adventure	1976	MGM / UA	PG
17	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
18	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
19	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
20	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
21	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
22	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
23	The Wizard of Oz	102	Adventure	1939	MGM / UA	G
24	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

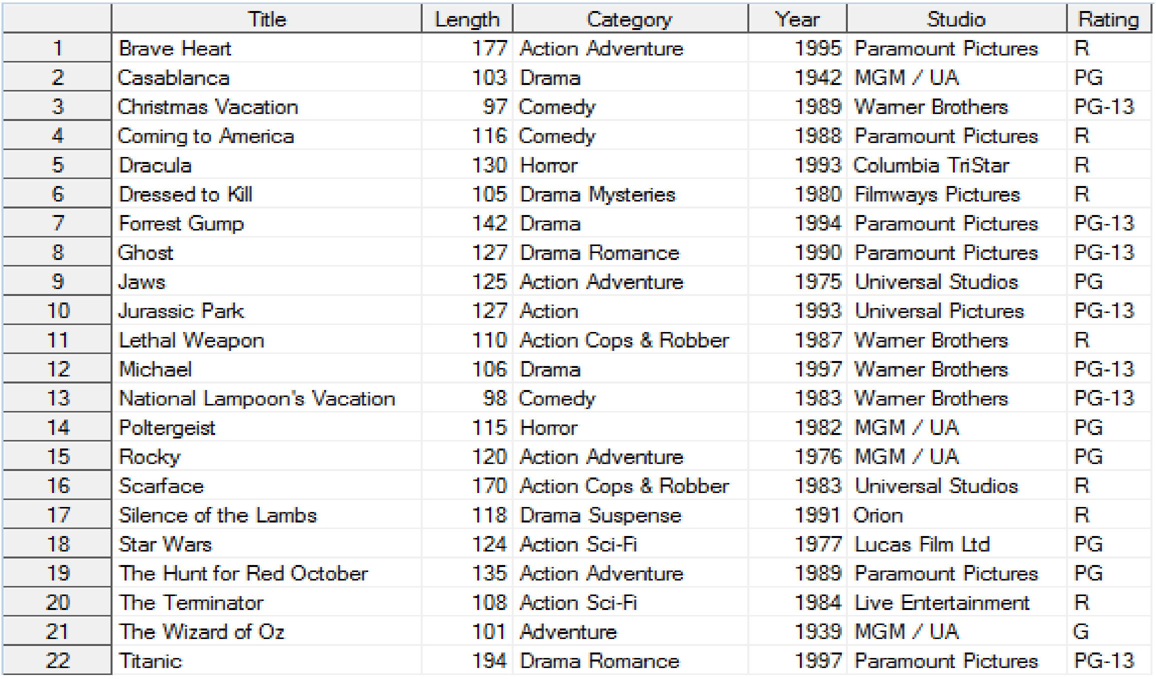
The NODUPKEYS (or NODUPKEY) Option

By specifying the **NODUPKEYS** (or **NODUPKEY**) option with PROC SORT, observations with duplicate keys are automatically removed from the output data set. The resulting output data set saw the removal of all the duplicate observations for Brave Heart, Forrest Gump, Rocky and The Wizard of Oz because they have duplicate keys data for the column, Title.

PROC SORT Code

```
PROC SORT DATA=Movies
           OUT=Movies_Sorted_without_DupKey
          NODUPKEYS ;
  BY Title ;
RUN ;
```

Resulting Table



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
9	Jaws	125	Action Adventure	1975	Universal Studios	PG
10	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
11	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
12	Michael	106	Drama	1997	Warner Brothers	PG-13
13	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
14	Poltergeist	115	Horror	1982	MGM / UA	PG
15	Rocky	120	Action Adventure	1976	MGM / UA	PG
16	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
17	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
18	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
19	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
20	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
21	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
22	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

<u>Note:</u> Although the removal of duplicates using PROC SORT is popular with many SAS users, an element of care should be given to using this method when processing big data sets. Because sort operations are time consuming and CPU-intensive operations, requiring as much as three times the amount of space to sort a data set, excessive demand is placed on system resources. Instead, SAS professionals may want to consider using PROC SUMMARY with the CLASS statement to avoid the need for sorting altogether, see Method #2.

Method #2 – Using PROC SQL to Remove Duplicates

The second method of removing duplicates uses PROC SQL. PROC SQL provides SAS users with an alternative to using PROC SORT, a particularly effective alternative for RDBMS users and SQL-centric organizations. Two approaches to removing duplicates will be illustrated, both using the **DISTINCT** keyword in a **SELECT** clause.

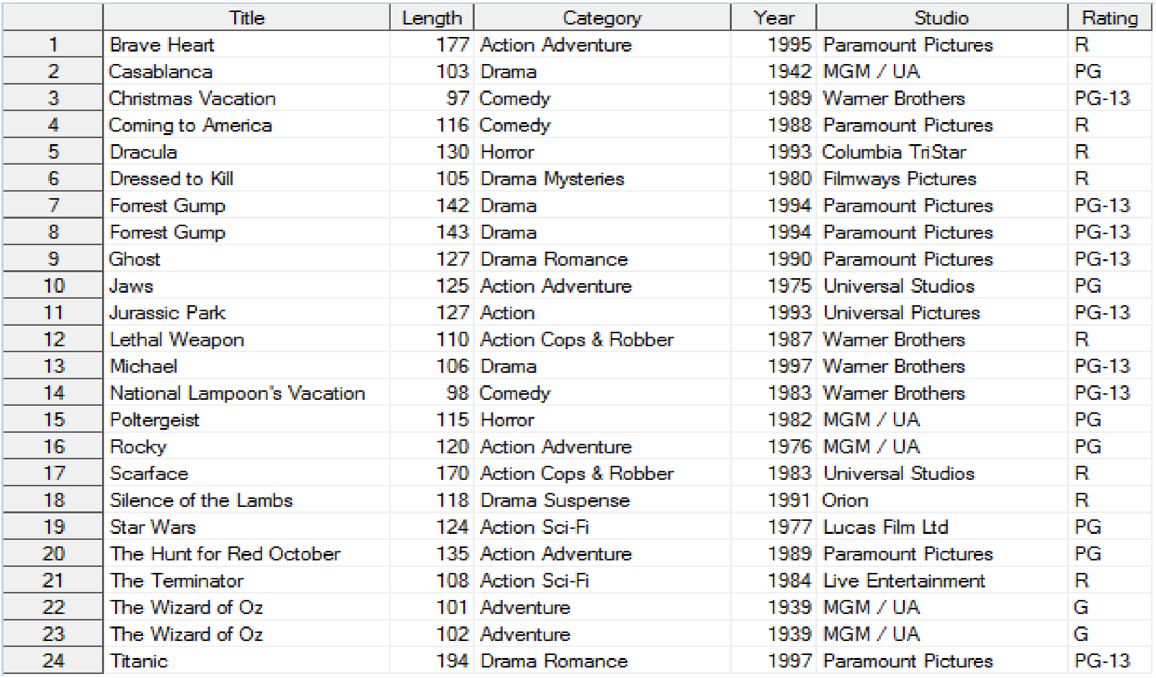
Specifying the DISTINCT Keyword

Using PROC SQL and the **DISTINCT** keyword provides SAS users with an effective way to remove duplicate rows where all the columns contain identical values. The following example removes duplicate rows using the DISTINCT keyword.

Removing Duplicate Rows using PROC SQL

```
proc sql ;
  create table Movies_without_DupRows as
    select DISTINCT (Title),
           Length,
           Category,
           Year,
           Studio,
           Rating
      from Movies_with_Dups
        order by Title ;
quit ;
```

Resulting Table



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Forrest Gump	143	Drama	1994	Paramount Pictures	PG-13
9	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
10	Jaws	125	Action Adventure	1975	Universal Studios	PG
11	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
12	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
13	Michael	106	Drama	1997	Warner Brothers	PG-13
14	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
15	Poltergeist	115	Horror	1982	MGM / UA	PG
16	Rocky	120	Action Adventure	1976	MGM / UA	PG
17	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
18	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
19	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
20	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
21	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
22	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
23	The Wizard of Oz	102	Adventure	1939	MGM / UA	G
24	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

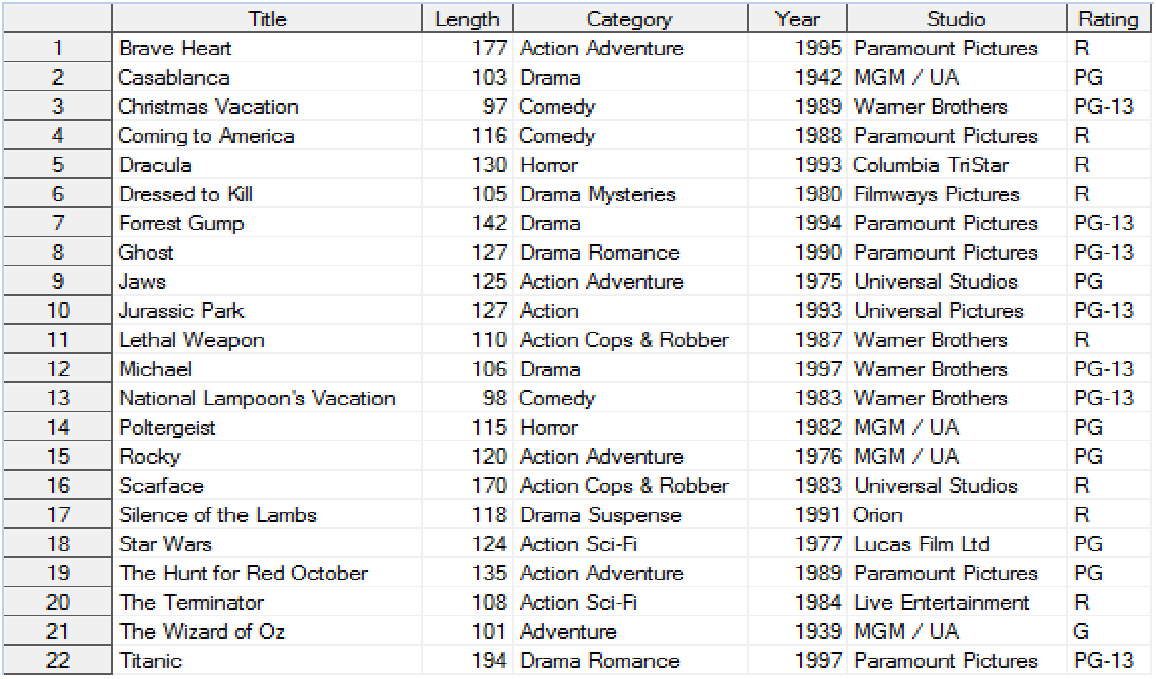
Specifying the DISTINCT Keyword, GROUP BY, HAVING-Clauses

Using the **DISTINCT** keyword, a **GROUP BY**-clause and **HAVING**-clause, rows with duplicate keys can be removed from an output table. The resulting output data set see the removal of all duplicate observations: Brave Heart, Forrest Gump, Rocky and The Wizard of Oz because they have duplicate keys data for the column, Title.

PROC SQL Code

```
proc sql ;
  create table work.Movies_without_DupKey as
    select DISTINCT(Title), Length, Category, Year, Studio, Rating
      from mydata.Movies_with_Dups
        group by Title
          having Title    = MAX(Title)
             AND Length   = MAX(Length)
             AND Category = MAX(Category)
             AND Year     = MAX(Year)
             AND Studio   = MAX(Studio)
             AND Rating   = MAX(Rating) ;
quit;
```

Resulting Table



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
9	Jaws	125	Action Adventure	1975	Universal Studios	PG
10	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
11	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
12	Michael	106	Drama	1997	Warner Brothers	PG-13
13	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
14	Poltergeist	115	Horror	1982	MGM / UA	PG
15	Rocky	120	Action Adventure	1976	MGM / UA	PG
16	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
17	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
18	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
19	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
20	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
21	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
22	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

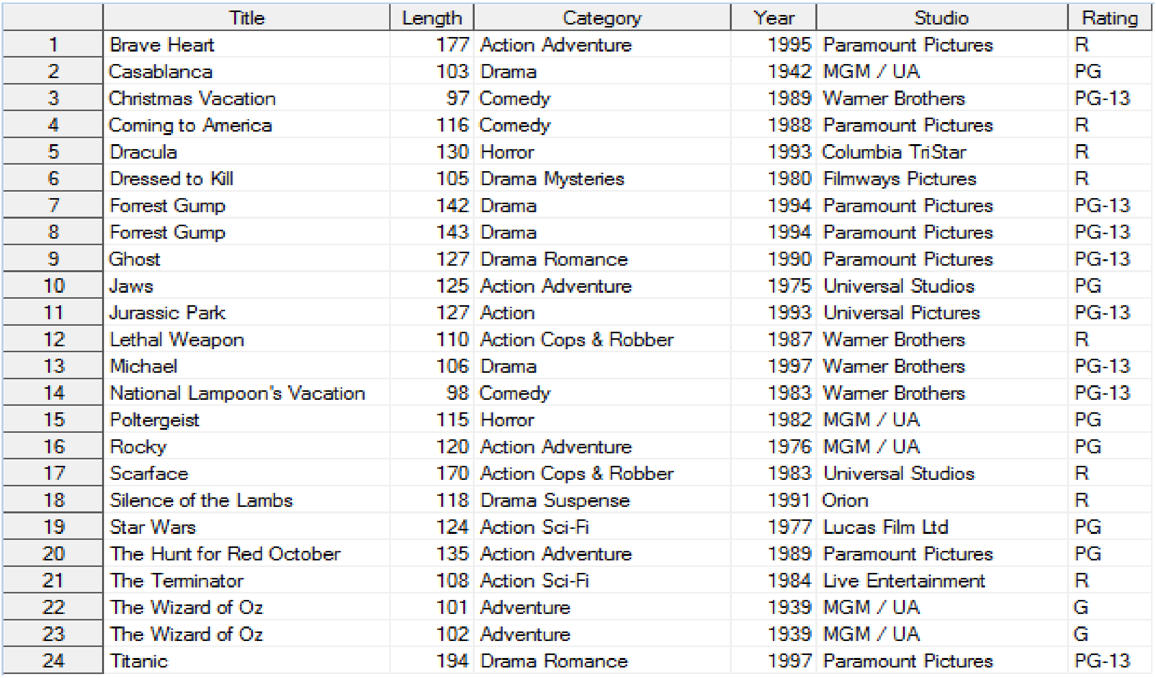
Method #3 – Using PROC SUMMARY to Remove Duplicates

The third method of removing duplicates uses PROC SUMMARY with the **CLASS** statement. Using PROC SUMMARY with the CLASS statement provides SAS professionals with a more efficient alternative than PROC SORT, and other methods, by avoiding the need for sorting in advance. Without the sorting requirement, considerably less system resources are needed to identify duplicates. But three additional aspects make this method an effective alternative: the specification of the **NWAY** parameter that corresponds to the combination of all CLASS variables, the specification of a **CLASS** statement to collapse observations with the same column values, and the creation of a **_FREQ_** column containing the number of occurrences. In the next example, a CLASS statement with all the variables is specified to select observations (rows) with multiple occurrences (duplicates) in the entire record (observation). The **OUTPUT OUT=** parameter renders the results to an output SAS data set.

Removing Rows with Duplicate Variable Values using PROC SUMMARY

```
proc summary data=Movies_with_Dups
             nway ;
  class Title Length Category Year Studio Rating ;
  id Length Category Year Studio Rating ;
  output out=Movies_Summary_without_DupRecs
            (drop=_type_) ;
run ;
proc print data=Movies_Summary_without_DupRecs
                (rename=(_freq_ = Dupkey))
                 noobs ;
run ;
```

Resulting Table



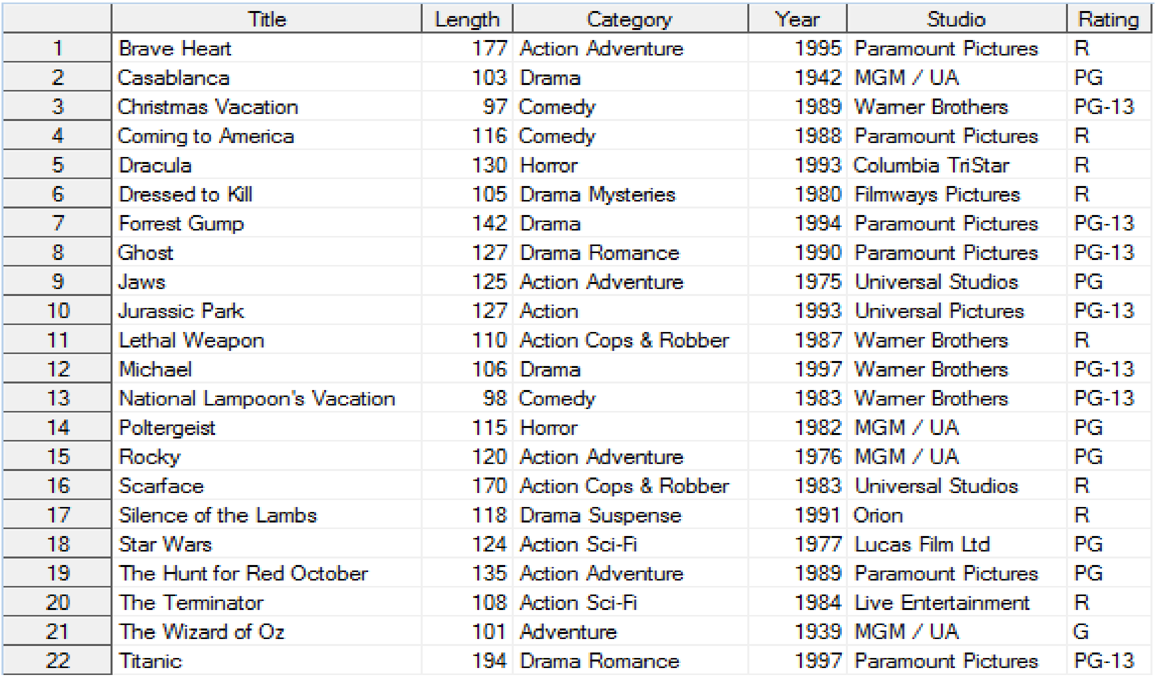
	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Forrest Gump	143	Drama	1994	Paramount Pictures	PG-13
9	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
10	Jaws	125	Action Adventure	1975	Universal Studios	PG
11	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
12	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
13	Michael	106	Drama	1997	Warner Brothers	PG-13
14	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
15	Poltergeist	115	Horror	1982	MGM / UA	PG
16	Rocky	120	Action Adventure	1976	MGM / UA	PG
17	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
18	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
19	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
20	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
21	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
22	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
23	The Wizard of Oz	102	Adventure	1939	MGM / UA	G
24	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

In the next example, a **CLASS** statement with the key variable is specified to select observations with multiple occurrences (duplicates) just in the key itself. The **OUTPUT OUT=** parameter renders the results to an output SAS data set.

Removing Rows with Duplicate Keys using PROC SUMMARY

```
proc summary data=Movies_with_Dups
             nway ;
  class Title ;
  id Length Category Year Studio Rating ;
  output out=Movies_Summary_without_DupKey
            (drop=_type_) ;
run ;
proc print data=Movies_Summary_without_DupKey
                (rename=(_freq_ = Dupkey))
                 noobs ;
run ;
```

Resulting Table



	Title	Length	Category	Year	Studio	Rating
1	Brave Heart	177	Action Adventure	1995	Paramount Pictures	R
2	Casablanca	103	Drama	1942	MGM / UA	PG
3	Christmas Vacation	97	Comedy	1989	Warner Brothers	PG-13
4	Coming to America	116	Comedy	1988	Paramount Pictures	R
5	Dracula	130	Horror	1993	Columbia TriStar	R
6	Dressed to Kill	105	Drama Mysteries	1980	Filmways Pictures	R
7	Forrest Gump	142	Drama	1994	Paramount Pictures	PG-13
8	Ghost	127	Drama Romance	1990	Paramount Pictures	PG-13
9	Jaws	125	Action Adventure	1975	Universal Studios	PG
10	Jurassic Park	127	Action	1993	Universal Pictures	PG-13
11	Lethal Weapon	110	Action Cops & Robber	1987	Warner Brothers	R
12	Michael	106	Drama	1997	Warner Brothers	PG-13
13	National Lampoon's Vacation	98	Comedy	1983	Warner Brothers	PG-13
14	Poltergeist	115	Horror	1982	MGM / UA	PG
15	Rocky	120	Action Adventure	1976	MGM / UA	PG
16	Scarface	170	Action Cops & Robber	1983	Universal Studios	R
17	Silence of the Lambs	118	Drama Suspense	1991	Orion	R
18	Star Wars	124	Action Sci-Fi	1977	Lucas Film Ltd	PG
19	The Hunt for Red October	135	Action Adventure	1989	Paramount Pictures	PG
20	The Terminator	108	Action Sci-Fi	1984	Live Entertainment	R
21	The Wizard of Oz	101	Adventure	1939	MGM / UA	G
22	Titanic	194	Drama Romance	1997	Paramount Pictures	PG-13

Conclusion

While many users use PROC SORT to remove duplicate observations (or rows) based on the key and/or the entire record from SAS data sets, two other approaches were shown. Since sorts can be expensive and time-consuming processes, it's advisable to use approaches that reduce the utilization of system resources to remove duplicates, such as with PROC SQL or PROC SUMMARY. A second approach to removing duplicates using PROC SQL was shown, because much of today's data resides in databases and a definite need to be able to use a universal language to remove duplicates exists. A final approach to removing duplicates using PROC SUMMARY and the CLASS statement was illustrated as a more efficient alternative to PROC SORT and PROC SQL, because it eliminates the need for sorting in advance.

References

Lafler, Kirk Paul. 2016. *Removing Duplicates Using SAS,* Proceedings of the Pharmaceutical SAS Users Group (PharmaSUG) Conference – 2016.

Acknowledgments

The author thanks David Corliss and Sherry Zhou, Tools of the Trade Section Chairs, for accepting my abstract and paper; Richann Watson, MWSUG 2016 Academic Chair; Adrian Katschke, MWSUG 2016 Operations Chair; the MidWest SAS Users Group (MWSUG) Executive Board; and SAS Institute for organizing and supporting a great conference!

Trademark Citations

About the Author

Kirk Paul Lafler is an entrepreneur, consultant and founder of Software Intelligence Corporation, and has been using SAS since 1979. Kirk is a SAS Certified Professional, provider of IT consulting services, advisor and professor at UC San Diego Extension and educator to SAS users around the world, mentor, and emeritus sasCommunity.org Advisory Board member. As the author of six books including Google® Search Complete (Odyssey Press. 2014) and PROC SQL: Beyond the Basics Using SAS, Second Edition (SAS Press. 2013); Kirk has written hundreds of papers and articles; been an Invited speaker and trainer at hundreds of SAS International, regional, special-interest, local, and in-house user group conferences and meetings; and is the recipient of 23 “Best” contributed paper, hands-on workshop (HOW), and poster awards.

Comments and suggestions can be sent to:

Kirk Paul Lafler
Senior SAS® Consultant, Application Developer, Data Analyst, Educator and Author
Software Intelligence Corporation
E-mail: KirkLafler@cs.com
LinkedIn: http://www.linkedin.com/in/KirkPaulLafler
Twitter: @sasNerd