

Paper 359-2013

How to Become a GTL / PROC TEMPLATE Jedi

Christopher Battiston, SickKids, Toronto, Ontario, Canada

ABSTRACT

This tongue-in-cheek paper will bring together Star Wars and SAS®, answering (at least potentially) how would SAS® have been used a long time ago in a galaxy far, far away? Using PROC TEMPLATE, GTL, and ODS, examples will be shown of reports that could have been used by the Empire and the Rebel Alliance. Topics will include creating reports for mobile devices, bringing in images into the reports, and creating dynamic reports - without using Jedi mind tricks on anyone!

INTRODUCTION

"A long, long ago in a galaxy far, far away..."

It is a period of analysis and graphics. Rebel SAS® programmers, writing code from a hidden base, have developed their first dashboard documenting their first victory against the evil Galactic Empire.

During deployment, Rebel analysts managed to hack into plans for the Empire's ultimate weapon, the DEATH STAR, an armoured space station with enough power to run analyses on datasets with trillions of rows and columns.

Pursued by the Empire's sinister Data Managers, Princess Leia races home aboard her starship, custodian of the stolen PROCs that can save her people and restore freedom to the galaxy..."



Figure 1. My Logo for this Paper

INTRODUCTION TO THE STAR WARS UNIVERSE

For those who are unfamiliar with the Star Wars story, a brief overview of the movies; please keep in mind that the movies cannot be done any justice in these short sentences but enough information will be provided to give an appreciation of the graphs.

A New Hope, the first film released in the series, was actually Episode IV (setting up for the prequels to be released the late 1990s) follows a group of freedom fighters known as the Rebel Alliance as they plot to destroy the powerful Death Star space station, a devastating weapon created by the evil Galactic Empire. This conflict disrupts the isolated life of farmboy Luke Skywalker when he inadvertently acquires the droids carrying the stolen plans to the Death Star. After the Empire begins a cruel and destructive search for the droids, Skywalker decides to accompany

Jedi Knight Obi-Wan Kenobi on a daring mission to rescue the owner of the droids, rebel leader Princess Leia, and save the galaxy.

The second film released, The Empire Strikes Back, was set three years after A New Hope. The Galactic Empire, under the leadership of the villainous Darth Vader, is in pursuit of Luke Skywalker and the rest of the Rebel Alliance. While Vader chases a small band of Luke's friends - Han Solo, Princess Leia, and others - across the galaxy, Luke studies the Force under Jedi Master Yoda. But when Vader captures Luke's friends, Luke must decide whether to complete his training and become a full Jedi Knight or to confront Vader and save his comrades.

Return of the Jedi, set one year later, has the evil Galactic Empire, under the direction of the ruthless Emperor, building a second Death Star in order to crush the Rebel Alliance. Since the Emperor plans to oversee the final stages of its construction, the Rebel Fleet launches a full-scale attack on the Death Star in order to prevent its completion and kill the Emperor, effectively bringing an end to the Empire. Meanwhile, Luke Skywalker, a Rebel Leader and Jedi Knight, struggles to bring Darth Vader back from the Dark Side of the Force.

The three more recent movies (Phantom Menace, The Attack of the Clones, and Revenge of the Sith) are my least favourite movies, and if you haven't seen any of the Star Wars movies I would recommend starting with A New Hope and moving through the order that they were released. The basic premise of the three prequels is the back story of Darth Vader, Obi-Wan Kenobi, and the Jedi.

SAS® AND STAR WARS – REALLY?

I've been a huge fan of the Star Wars universe ever since my dad took me to see Empire Strikes Back in theatres, back in 1980. I was 4 at the time and I had no idea that this was something that I'd be carrying with me for the rest of my life; yet here I am, still doing the marathon at least twice a year, still getting into debates with friends (Han Solo did not shoot first in the original production of A New Hope!) and now, writing a paper about SAS® and Star Wars! I wanted to do something fun and light-hearted and thought that this was something that could both educate people on PROC TEMPLATE and GTL and provide a break from all the other hardcore SAS® presentations.

SO WHAT CAN I POSSIBLY BE PRESENTING?

I'm going to go through PROC TEMPLATE, GTL and ODS Graphics and how they would have been used if SAS® was available in the Star Wars universe. Along with that, I have a couple of graphs about the movies based on my likes / dislikes, and finally a deviation from the Graphics side to a run-through of other SAS-specific features available in Star Wars.

GRAPHS BASED ON THE MOVIES

NUMBER OF STROMTROOPERS ORDERED, RECEIVED AND DESTROYED BY THE REBELS

One of the ubiquitous yet not really developed groups of characters in the original Star Wars movies are the Stormtroopers – the soldiers on the side of the Empire, they are sort of the sufferers of the “Red Shirt Syndrome” from Star Trek – for a group of supposedly highly trained military fighters, they die an awful lot and don't really accomplish any real goals. So, I would have to assume that the Empire has someone that is tracking and monitoring how many are ordered, received, and destroyed (in the prequels it is revealed that they are in fact Clones, and that a civilization is responsible for their creation and training).

Obs	year	ordered	received	destroyed
1	ABY0	1.2	1.0	501235
2	ABY1	2.7	2.2	784956
3	ABY2	3.5	3.9	659775
4	ABY3	1.7	1.6	702548
5	ABY4	3.6	3.6	2051486

Table 1. Stormtrooper Turnaround by Year

For those unfamiliar with the movies, in the Return of the Jedi the Empire is completely destroyed – which is why I've got the outlier at the top right of the third panel. The ABY is the way that they indicate years, and means “After the Battle of Yavin” (not actually in the movies).

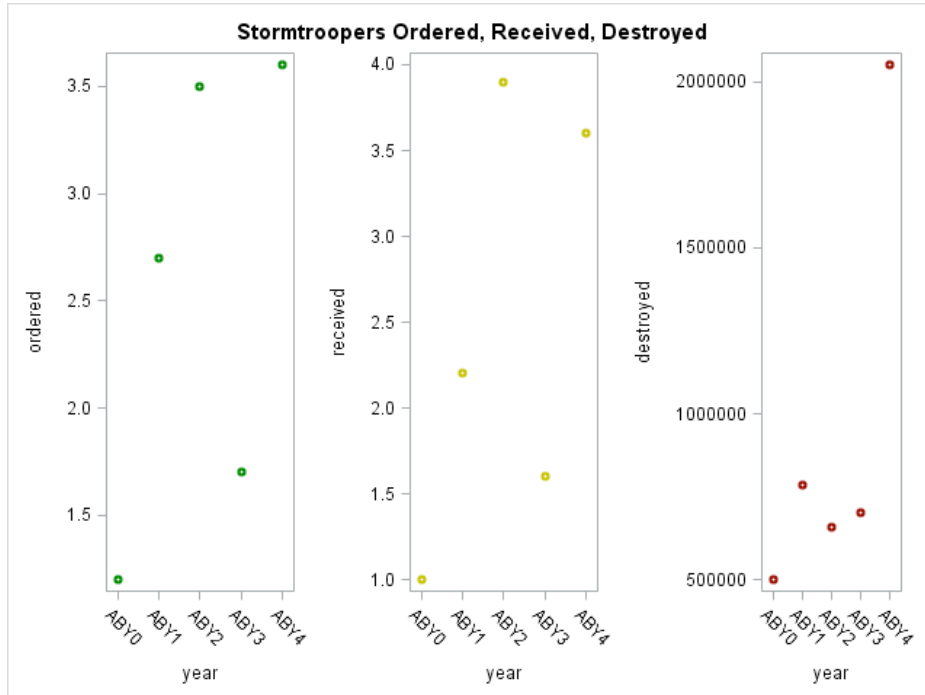


Figure 2. Stormtrooper Turnaround Plotted

Here's the code I used, including the data step to create my table. I've highlighted a couple of key pieces of code.

```

DATA stormtroopers;
input year $4. ordered received destroyed;

datalines ;
ABY0 1.2 1 501235
ABY1 2.7 2.2 784956
ABY2 3.5 3.9 659775
ABY3 1.7 1.6 702548
ABY4 3.6 3.6 2051486
;
proc print data=stormtroopers;
run;
proc template;
define statgraph stormtrooper;
begingraph;
entrytitle "Stormtroopers Ordered, Received, Destroyed";
layout lattice / rows=1 columns=3 rowgutter=10 columngutter=10;
layout overlay;
scatterplot x=year y=ordered / markerattrs=(color=green weight=bold);
endlayout;
layout overlay;
scatterplot x=year y=received / markerattrs=(color=darkyellow
weight=bold);
endlayout;
layout overlay;
scatterplot x=year y=destroyed / markerattrs=(color=darkred weight=bold);
endlayout;
endgraph;
end;
run;
proc sgrender data=stormtroopers template=stormtrooper;
run;

```

Because I've created the graph as a side-by-side scatter plot, you see the "layout lattice / rows=1 columns=3" in the PROC TEMPLATE. This tells SAS® that we want a 1 x 3 display, and as long as your rows x columns matches the number of graphs being generated, you can have any arrangement that suits your needs.

You will also notice that there are "markerattrs" options in each of the scatterplot statements; I've kept them relatively simple by changing the weight and colour, but you can get very complex with these.

DEATH STAR RESOURCES

There were two Death Stars; the first one, in A New Hope, is reported to have been about 161 km (100 miles) in diameter. There were 265,675 crew, as well as 52,276 gunners, 607,360 troops, 30,984 stormtroopers, 42,782 ship support staff, and 180,216 pilots and support crew. So there were a total of 1,179,293 people in a moon-sized base, and that must have been absolutely overwhelming to maintain, clean and operate. (I remember being a kid and watching one of the movies that showed the inside of the Death Star and wondered how many people it took to clean all those corridors, rooms, etc. etc.). So, here in millions of Galactic Credits, would be what could have been spent on different areas and the staff and equipment for those areas.

Area	Staff Admin	Staff Stormtroopers	Staff Maint	Staff Housekeeping	Equip Capital	Equip Expense	Technology
Science_RnD	4	13	15	6	15	15	6
Military_RnD	13	3	15	2	15	15	5
LivingQuarters	14	9	2	15	2	9	5
HR	3	15	1	12	6	5	12
Military	10	6	7	3	12	7	5
Science	13	7	5	3	11	13	6
Agriculture	14	15	4	2	7	5	11
Infrastructure	6	6	7	7	8	13	9
Waste_Reclamation	12	5	2	14	13	6	2
Food_Services	13	10	2	4	13	5	12
Info_Services	6	8	9	5	10	5	4
Info_Technology	13	6	9	8	5	9	11
Communications	12	4	11	8	5	3	12
TOTAL	133	107	89	89	122	110	100

Table 2. Death Star Resources by Area and Cost Type

Obs	CostType	Total
1	Staff_Admin	133
2	Staff_Stormtroopers	107
3	Staff_Maint	89
4	Staff_Housekeeping	89
5	Equip_Capital	122
6	Equip_Expense	110
7	Technology	100

Table 3. Totals by Cost Type

The first graph just shows a basic plot of the areas and their administrative costs; the purpose of this is to highlight the use of the Horizontal Reference line at 11.7.

The second graph shows just two of the groups, as a way of showing overlapping bar charts. In the third screenshot, a simple bar graph is sufficient to highlight the totals, and clearly shows that the majority of costs are put towards the Administrative Staff.

The last graph was not actually generated by PROC TEMPLATE but rather by PROC GCHART, and is intended to show that although PROC TEMPLATE is very powerful, I could not get the image to import and so had to resort to an older PROC; however, the results are effective and what I wanted to accomplish.

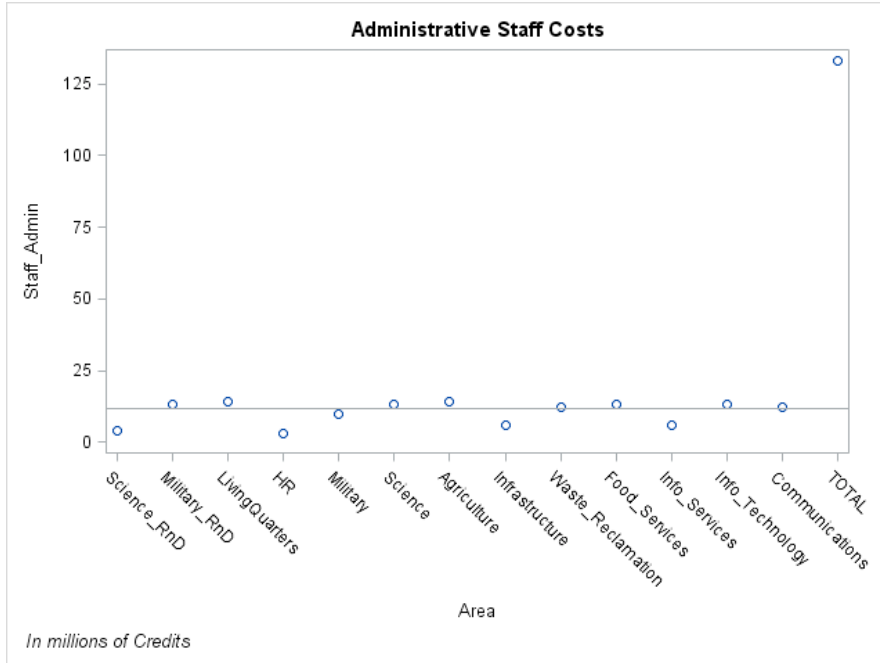


Figure 3. Administrative Staff Costs (use of the HREF line)

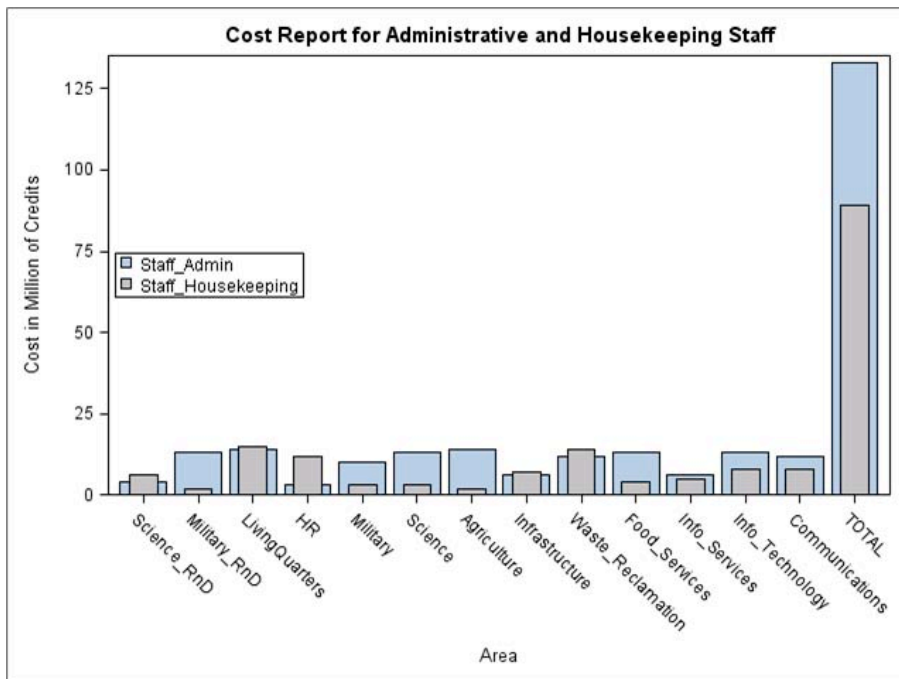


Figure 4. Cost Report for Administrative and Housekeeping Staff (use of Bar-in-Bar Graph and Legend)

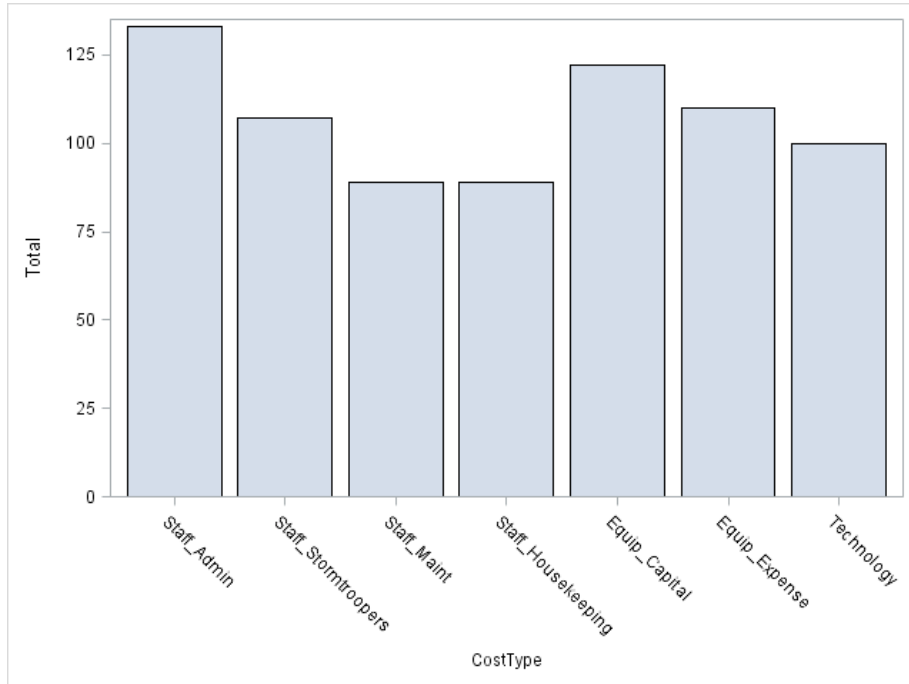


Figure 5. A Simple Bar Chart for Total Costs



Figure 6a. The Logo of the Galactic Empire

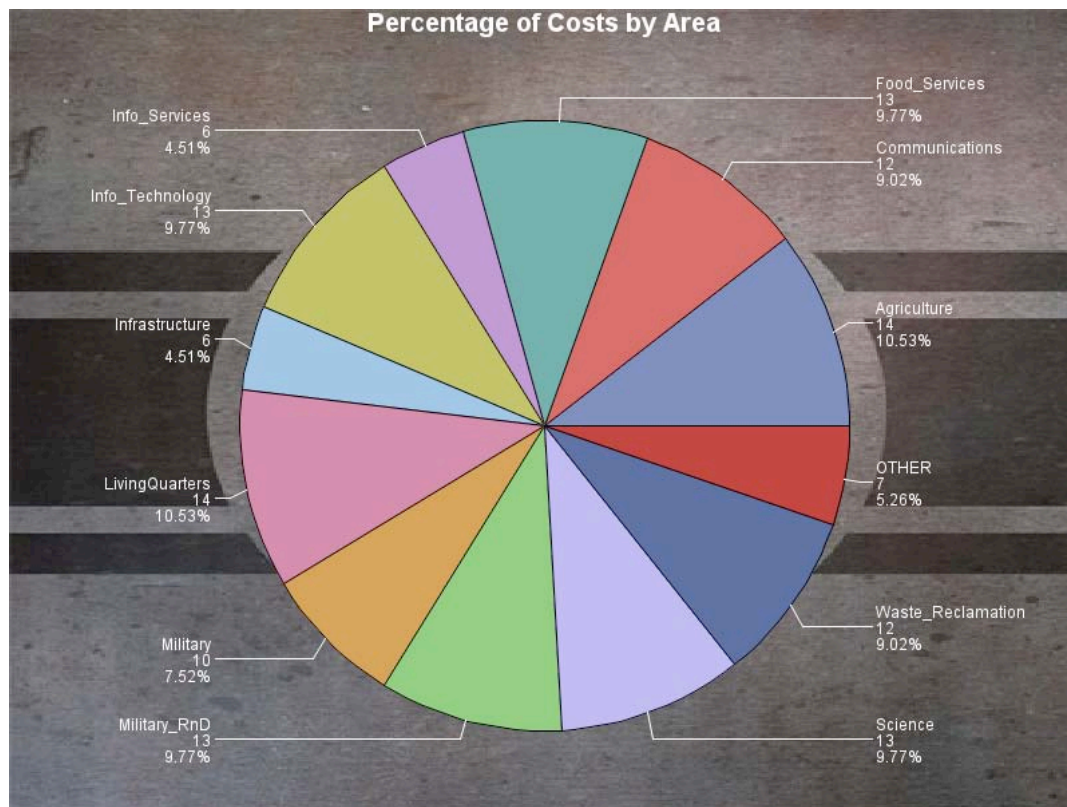


Figure 6b. The Total Costs as a Pie Chart with the Logo as the Background

```

DATA WORK.DEATHSTAR;
INPUT
    Area $1-17 Staff_Admin Staff_Stormtroopers Staff_Maint Staff_Housekeeping
    Equip_Capital Equip_Expense Technology;
DATALINES;
Science_RnD      4 13 15 6 15 15 6
Military_RnD     13 3 15 2 15 15 5
LivingQuarters   14 9 2 15 2 9 5
HR               3 15 1 12 6 5 12
Military         10 6 7 3 12 7 5
Science         13 7 5 3 11 13 6
Agriculture     14 15 4 2 7 5 11
Infrastructure   6 6 7 7 8 13 9
Waste_Reclamation 12 5 2 14 13 6 2
Food_Services   13 10 2 4 13 5 12
Info_Services   6 8 9 5 10 5 4
Info_Technology 13 6 9 8 5 9 11
Communications  12 4 11 8 5 3 12
TOTAL           133 107 89 89 122 110 100
;

proc print data=deathstar;
run;

DATA WORK.DEATHSTAR2;
INPUT
    CostType $19. Total;
Datalines;
Staff_Admin      133
Staff_Stormtroopers 107
Staff_Maint      89
    
```

```

Staff_Housekeeping 89
Equip_Capital      122
Equip_Expense      110
Technology          100
;

proc print data=deathstar2;
run;
/* Graph One - Plot with Horizontal Reference Line */
proc template;
define statgraph sgdesign;
dynamic _AREA _STAFF_ADMIN;
begingraph;
  entrytitle halign=center 'Administrative Staff Costs';
  entryfootnote halign=left 'In millions of Credits';
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10
columnngutter=10;
  layout overlay;
    scatterplot x=_AREA y=_STAFF_ADMIN / name='scatter';
    referenceline y=11.7 / name='href' yaxis=Y curvelabelposition=max;
  endlayout;
endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.DEATHSTAR (wheretemplate=sgdesign;
dynamic _AREA="AREA" _STAFF_ADMIN="STAFF_ADMIN'n";
run;

/* Graph Two - Bar-in-Bar Chart */
proc template;
define statgraph sgdesign;
dynamic _AREA _STAFF_ADMIN _AREA2 _STAFF_HOUSEKEEPING;
begingraph;
  entrytitle halign=center 'Cost Report for Administrative and Housekeeping
Staff';
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10
columnngutter=10;
  layout overlay / yaxisopts=( label=('Cost in Million of Credits'));
  barchart x=_AREA y=_STAFF_ADMIN / name='bar' clusterwidth=1.0;
  barchart x=_AREA2 y=_STAFF_HOUSEKEEPING / name='bar2' barwidth=0.51
groupdisplay=Cluster clusterwidth=1.0 fillattrs=(color=CXC6C3C6 );
  discretelegend 'bar' 'bar2' / opaque=false border=true halign=left
valign=center displayclipped=true across=1 order=rowmajor location=inside;
  endlayout;
endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.DEATHSTAR template=sgdesign;
dynamic _AREA="AREA" _STAFF_ADMIN="STAFF_ADMIN'n" _AREA2="AREA"
_STAFF_HOUSEKEEPING="STAFF_HOUSEKEEPING'n";
run;

/* Graph Three - Simple Bar Chart */
proc template;
define statgraph Graph;
dynamic _COSTTYPE _TOTAL;
begingraph;
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10
columnngutter=10;

```



```

        layout overlay / xaxisopts=( labelattrs=(family='Franklin Gothic Medium'
));
        barchart x=_COSTTYPE y=_TOTAL / name='bar' discreteoffset=0.0
groupdisplay=Cluster clusterwidth=1.0;
        endlayout;
        endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.DEATHSTAR2 template=Graph;
dynamic _COSTTYPE="COSTTYPE" _TOTAL="TOTAL";
run;

/* Graph Four - Addition of an Image to a Graph */

goptions iback="d:/empire.jpg" imagestyle=fit ;
title1 color=white "Percentage of Costs by Area" ;
proc gchart data=work.deathstar ;
pie area / noheading
        freq=staff_admin
        percent=arrow
        plabel=(color=white)
;
where area<>'TOTAL';
run;
quit;

```

In the code for Graph One, I've highlighted the line that generates the reference line; the remainder of the code is more or less straightforward. For Graph Two, the two different portions that generate the bar graphs as well as the `barwidth` and the `location` showing that this will be the inner portion of the bar-in-bar. For Graph Three, it is as simple as one can get but I have changed the X-axis font to Franklin Gothic Medium just to show how that's done. The remainder of the code for Graph Three is also basic PROC TEMPLATE. The fourth Graph, which is not PROC TEMPLATE or GTL, uses the `iback` option to import the image EMPIRE.JPG. Using the `IMAGESTYLE=fit` enables the image to fit within the boundaries of the graph; left alone, the image would be tiled. I've also changed the colour of the title and labels (using `color=white` and `plabel=(color=white)`) so that they are clearer on the dark background. Finally, the `where` statement was added to exclude the overall total, as that would skew the pie chart unnecessarily.

REBEL ALLIANCE WHO ARE OUR FRIENDS?

Particularly in the prequels, there are frequent comments being made about systems or planets joining one side or the other. Someone from the Rebel Alliance had to be monitoring this so they would know where friendly planets were, where they could go to for help, and where to avoid. Here's a table detailing the 4 year span between "A New Hope" and "Return of the Jedi" for three sectors.

Obs	year	sector	allied	na	empire
1	ABY0	1	12	15	32
2	ABY1	1	16	2	27
3	ABY2	1	5	55	4
4	ABY3	1	7	5	9
5	ABY4	1	12	17	55
6	ABY0	2	82	12	1
7	ABY1	2	32	56	1
8	ABY2	2	25	27	19
9	ABY3	2	15	6	16
10	ABY4	2	16	16	21
11	ABY0	3	15	86	5
12	ABY1	3	77	12	2
13	ABY2	3	1	1	87
14	ABY3	3	12	3	92
15	ABY4	3	11	15	15

Table 4. Allied Planets with The Rebels

And below is the multi-level graph plotting these details; the panels are the Allies, Empire and Not Available, and then the lines within each panel are the three different sectors.

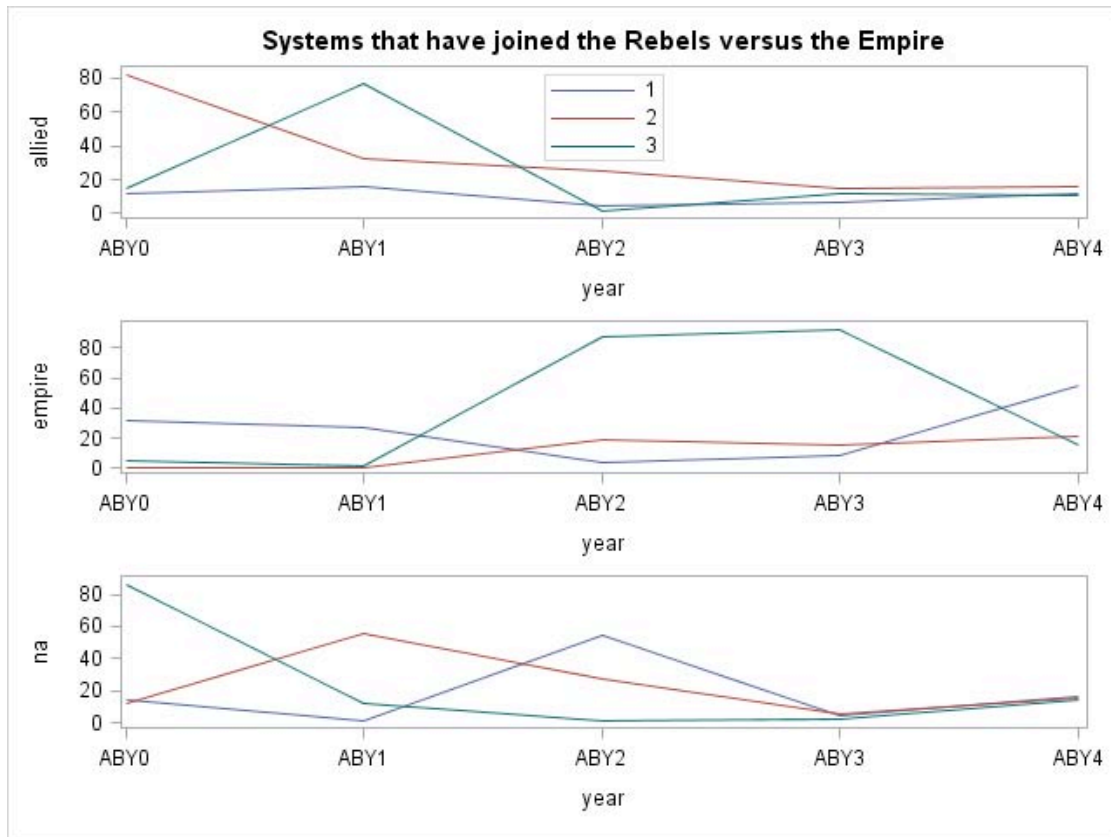


Figure 7. Three-tier Graph of Systems that have joined the Rebels

Here is the data to create the table, as well as the corresponding graphs.

DATA rebels;

input year \$4. sector allied na empire;

datalines ;

ABY0 1 12 15 32

ABY1 1 16 2 27

ABY2 1 5 55 4

ABY3 1 7 5 9

ABY4 1 12 17 55

ABY0 2 82 12 1

ABY1 2 32 56 1

ABY2 2 25 27 19

ABY3 2 15 6 16

ABY4 2 16 16 21

ABY0 3 15 86 5

ABY1 3 77 12 2

ABY2 3 1 1 87

ABY3 3 12 3 92

ABY4 3 11 15 15

;

proc print data=rebels;

run;

proc template;

define statgraph sgdesign;

dynamic _YEAR _ALLIED _SECTOR _YEAR2 _EMPIRE _SECTOR2 _YEAR3 _NA _SECTOR3;

beginningraph;

entrytitle halign=center 'Systems that have joined the Rebels versus the Empire';

```

entryfootnote halign=left ";
layout lattice / rowdatarange=data columndatarange=data rows=3 rowgutter=10 columngutter=10;
  layout overlay;
    seriesplot x=_YEAR y=_ALLIED / group=_SECTOR name='series' connectorder=xaxis;
    discretelegend 'series' / opaque=false border=true halign=center valign=top displayclipped=true across=1
order=rowmajor location=inside;
  endlayout;
  layout overlay;
    seriesplot x=_YEAR2 y=_EMPIRE / group=_SECTOR2 name='series2' connectorder=xaxis;
  endlayout;
  layout overlay;
    seriesplot x=_YEAR3 y=_NA / group=_SECTOR3 name='series3' connectorder=xaxis;
  endlayout;
endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.REBELS template=sgdesign;
dynamic _YEAR="YEAR" _ALLIED="ALLIED" _SECTOR="SECTOR" _YEAR2="YEAR" _EMPIRE="EMPIRE"
_SECTOR2="SECTOR" _YEAR3="YEAR" _NA="NA" _SECTOR3="SECTOR";
run;

```

The key here is the `layout lattice` and the `rows` parameters. These are telling SAS® to put it all together in stacked rows; you can do the same thing with `columns`, if you wanted them side-by-side.

ENERGY USAGE OF THE MILLENIUM FALCON

The Millennium Falcon, a spacecraft used by Han Solo and his co-pilot Chewbacca, was designed for smuggling cargo; as a result, it had certain “additions” made by Han and Chewbacca that allowed it to go to Hyperdrive. Because of these changes, I assumed that there would have to be reports generated to see how efficient the changes were. Here’s a partial output from that report:

Obs	period	distance	cargo	usage
1	1	2.0	Y	220
2	2	1.7	N	223
3	3	12.3	N	2569
4	4	0.9	Y	107
5	5	15.0	N	1625
6	6	22.0	Y	2500
7	7	14.6	N	1588
8	8	15.0	Y	1720
9	9	4.5	Y	501

Table 5. Energy Usage of the Falcon

Once again, the corresponding graphs are below. Because of the Falcon being used as a smuggling ship, I wanted to see if there was any correlation between Energy Usage and whether or not cargo was on board. However, these were obviously created by someone that doesn’t really understand effective graphs because of the combining of different chart types, x-axes not lined up, and different scales on the Y-axes. However, it does demonstrate what is

possible using PROC Template and GTL.

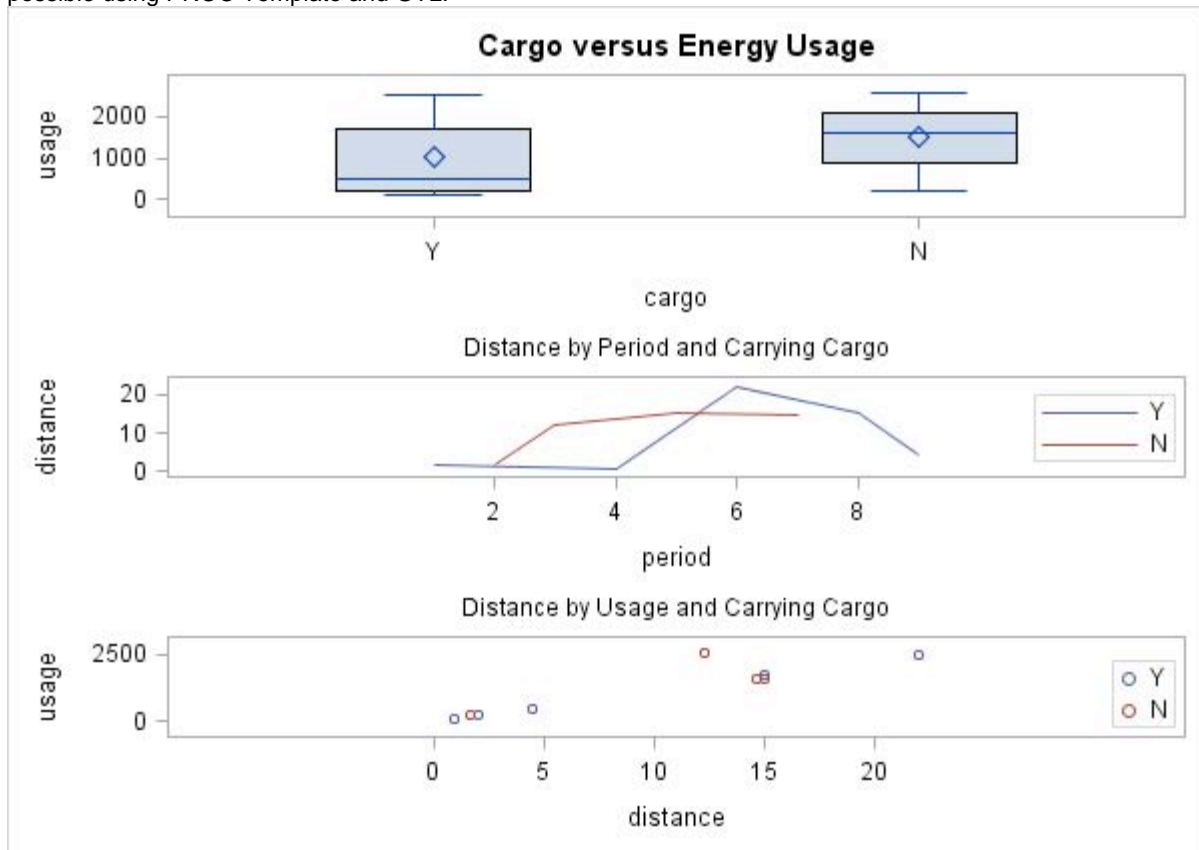


Figure 8. Graphs for the Falcon's Energy Usage and Cargo

And here's the PROC TEMPLATE and the DATA step to create the table:

```
data falcon;
input period distance cargo $1. usage;
datalines;
1 2 Y 220
2 1.7 N 223
3 12.3 N 2569
4 0.9 Y 107
5 15 N 1625
6 22 Y 2500
7 14.6 N 1588
8 15 Y 1720
9 4.5 Y 501
;

proc print data=falcon;
run;

proc template;
define statgraph sgdesign;
dynamic _USAGE _CARGO _PERIOD _DISTANCE _CARGO2 _DISTANCE2 _USAGE2 _CARGO3;
begingraph / designwidth=600 designheight=425;
  entrytitle halign=center 'Cargo versus Energy Usage';
  layout lattice / rowdatarange=data columndatarange=data rows=3 rowgutter=10
  columngutter=10;
  layout overlay;
  boxplot x=_CARGO y=_USAGE / name='box' boxwidth=0.4 clusterwidth=1.0;
endlayout;
enddefine;
```

```

    layout overlay;
        seriesplot x=_PERIOD y=_DISTANCE / group=_CARGO2 name='series'
connectorder=xaxis;
        discretelegend 'series' / opaque=false border=true halign=right valign=center
displayclipped=true across=1 order=rowmajor location=inside;
        entry halign=center 'Distance by Period and Carrying Cargo' / valign=top
location=outside;
    endlayout;
    layout overlay;
        scatterplot x=_DISTANCE2 y=_USAGE2 / group=_CARGO3 name='scatter';
        discretelegend 'scatter' / opaque=false border=true halign=right
valign=bottom displayclipped=true across=1 order=rowmajor location=inside;
        entry halign=center 'Distance by Usage and Carrying Cargo' / valign=top
location=outside;
    endlayout;
    endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.FALCON template=sgdesign;
dynamic _USAGE="USAGE" _CARGO="CARGO" _PERIOD="PERIOD" _DISTANCE="DISTANCE"
_CARGO2="CARGO" _DISTANCE2="DISTANCE" _USAGE2="USAGE" _CARGO3="CARGO";
run;

```

You can once again see the `layout lattice` and the `rows` parameters being set. However, you can also see the three distinct graphs being specified and the options including the `discretelegend`.

GRAPH ABOUT THE MOVIES

It was my intention to pull at least one marathon while writing this paper so that I could come up with other graphs, but about the movies themselves. However, finding 17 hours of free time is a lot to ask and so I have had to forego ones like frequency of JarJar Binks being annoying, Effectiveness of Clones in the battle against the Gungans, and time-to-event analysis of Anakin Skywalker complaining that Obi Wan is being unfair. So, as a fallback, I've come up with my "Coolness Factor by Movie" plot.

Any hardcore Star Wars fan has very strong opinions about which is the "best" movie, and I'm no exception. I wanted to see how I compared with Rotten Tomatoes and IMDB for all six movies. I find it interesting we are very comparable with the exception of "Revenge of the Sith", which in my opinion was *horrible* – but as it's part of the series, I still watch it as part of my marathon.

Here's the data table with the scores:

Obs	year	title	rottentomatoes	imdb	myscore
1	1977	A New Hope	9.3	8.8	8.5
2	1980	Empire Strikes Back	9.7	8.8	9.8
3	1983	Return of the Jedi	7.9	8.4	8.1
4	1999	The Phantom Menace	5.7	6.5	6.7
5	2002	Attack of the Clones	6.7	6.7	6.0
6	2005	Revenge of the Sith	8.0	7.7	0.1

Table 6. Ratings of the 6 Star Wars Movies

And here is the graph; to spice things up a little bit I decided to see what sort of unique markers I could use, and so I went with TILDE, ASTERISK and UNION. I put the legend in the bottom left, just so it was out of the way of the markers.

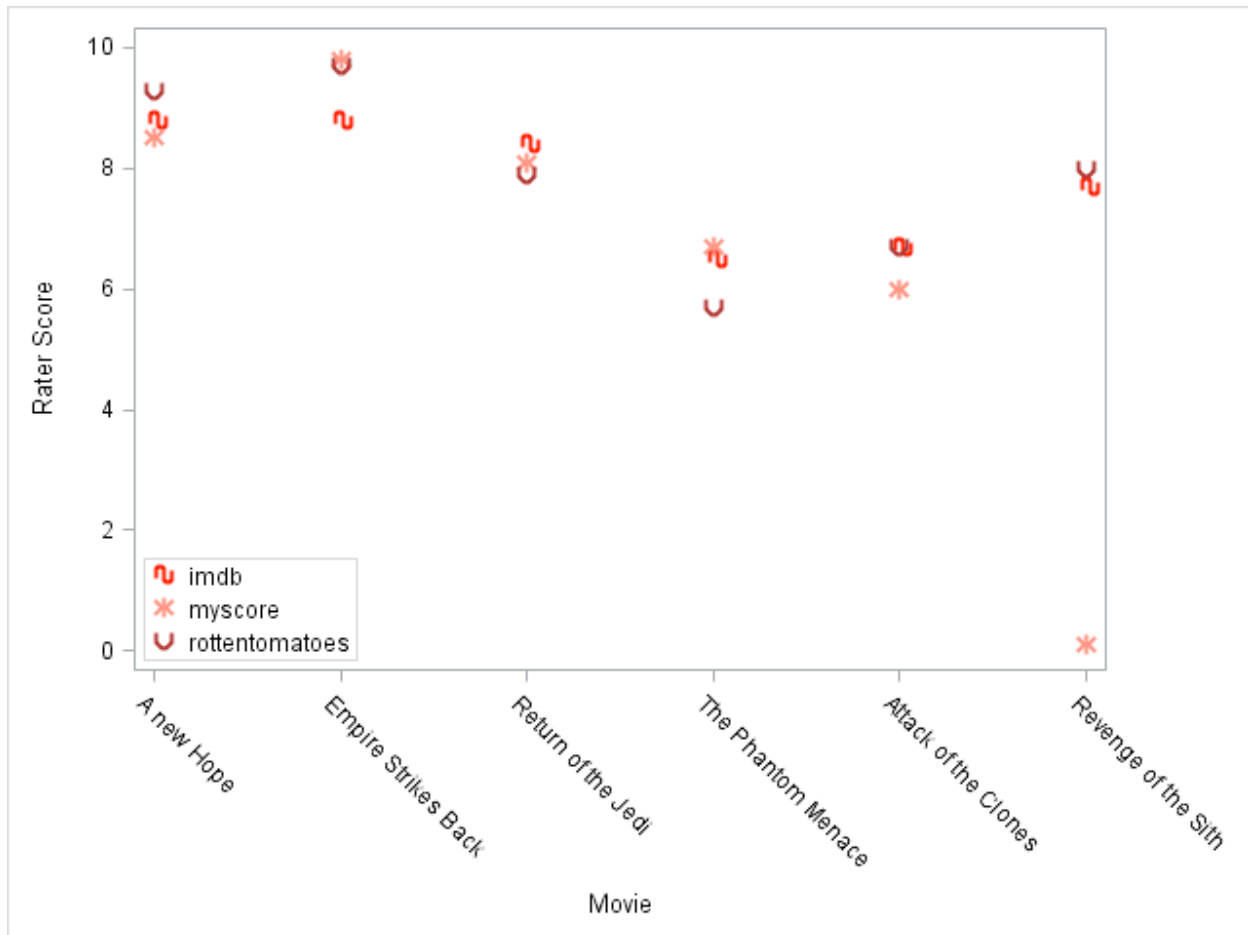


Figure 9. The Ratings Plotted

And once again, the code used to generate the table and the graphs. Because I'm new to changing marker colour, I wanted to use the Hexadecimal names (Red, Peach, and Dark Red) to become accustomed to searching for and using them.

```
DATA movies;
input year $4. title $21. rottentomatoes imdb myscore;

datalines ;
1977 A new Hope          9.3 8.8 8.5
1980 Empire Strikes Back 9.7 8.8 9.8
1983 Return of the Jedi  7.9 8.4 8.1
1999 The Phantom Menace  5.7 6.5 6.7
2002 Attack of the Clones 6.7 6.7 6.0
2005 Revenge of the Sith 8.0 7.7 0.1
;

proc print data=movies;
run;

proc template;
define statgraph Graph;
dynamic _TITLE _IMDB _TITLE2 _MYSCORE _TITLE3 _ROTTENTOMATOES;
begingraph;
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10
columngutter=10;
  layout overlay / xaxisopts=( label=('Movie')) yaxisopts=( label=('Rater
Score'));
  scatterplot x=_TITLE y=_IMDB / name='scatter' discreteoffset=0.02
```

```

markerattrs=(color=CXFF0000 symbol=TILDE size=10 weight=bold );
scatterplot x=_TITLE2 y=_MYSCORE / name='scatter2'
markerattrs=(color=CXFF8273 symbol=ASTERISK size=10 weight=bold );
scatterplot x=_TITLE3 y=_ROTTENTOMATOES / name='scatter3'
markerattrs=(color=CXA52829 symbol=UNION size=10 weight=bold );
discretelegend 'scatter' 'scatter2' 'scatter3' / opaque=false border=true
halign=left valign=bottom displayclipped=true across=1 order=rowmajor location=inside;
endlayout;
endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.MOVIES template=Graph;
dynamic _TITLE="TITLE" _IMDB="IMDB" _TITLE2="TITLE" _MYSCORE="MYSCORE" _TITLE3="TITLE"
_ROTTENTOMATOES="ROTTENTOMATOES" ;
run;

```

The key lines here are `markerattrs` which allows you full flexibility over the specifications of your markers. This is really nice if you have company colours that you want to use, combined with the company logo in the background as in the pie chart in the previous section, would make an impactful presentation.

NEED TO BE MOBILE TO AVOID THE EMPIRE!

Because the Rebel Alliance is constantly on the move, they have to rely on being able to quickly send out reports to the various squadrons, allies, and leaders so that everyone can have as “real time” data as possible on their mobile devices or computers.. Again this does not use the Template or GTL processes, but because it fit into my theme of this paper, and is something I’m now using in my regular day job, I wanted to have it as part of this paper. Here’s the screen shot of the email and embedded graph on my iPhone 4S; as with everything else iDevice, zooming in and out is enabled so you can focus in on a particular area with ease.

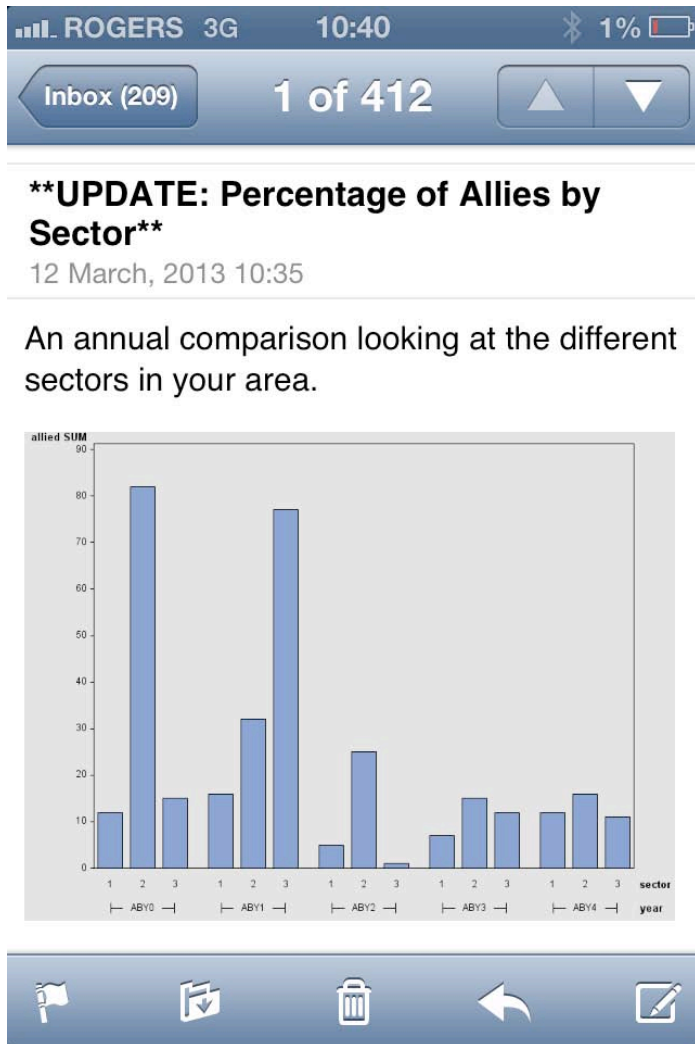


Figure 10. Output of the Graphs to my iPhone 4S

Here's the code I used to generate this graph, using the WORK.REBELS dataset that was created above.

```
filename grafout 'd:\temp\Allies.gif';
options reset=all device=gif gsfname=grafout gsfmode=replace;
axis1 minor=none;
```

```
proc gchart data=work.rebels;
  vbar sector / sumvar=allied group=year
           nolegend discrete width=10 space=3
           raxis=axis1;
run;
```

```
quit;
```

```
filename grafout clear;
filename sendmail email "my.email@work.ca"
  subject = "**UPDATE: Percentage of Allies by Sector**"
  attach = "d:\temp\Allies.gif";
```

```
data _null_;
  file sendmail;
  put "An annual comparison looking at the different sectors in your area.";
run;
```

```
filename sendmail clear;
```

As you can see I'm specifying my out file in the first line, `device=gif`, and then a DATA step is used to send the email. **This was only tested on an iPad and an iPhone.** I cannot guarantee Blackberries etc. will behave the same way.

IF SAS® CUSTOMISED PROCS FOR STAR WARS

Because SAS® has the best customer service, it would provide customized PROCs to the Empire (because you can't say no to Darth Vader!) and to the Rebels (because you have to do what you can to help the good guys!) as part of their licensing agreement.

PROC TAUNTON

Tauntons are hairy, two-legged beasts that live on the ice world of Hoth. Both Han Solo and Luke Skywalker had unfortunate events happen with theirs, and so I figured their track record must be pretty poor. PROC TAUNTON therefore would be a survival analysis of the Tautons, and in this example I'm comparing Luke, Han and No One riding the beast.

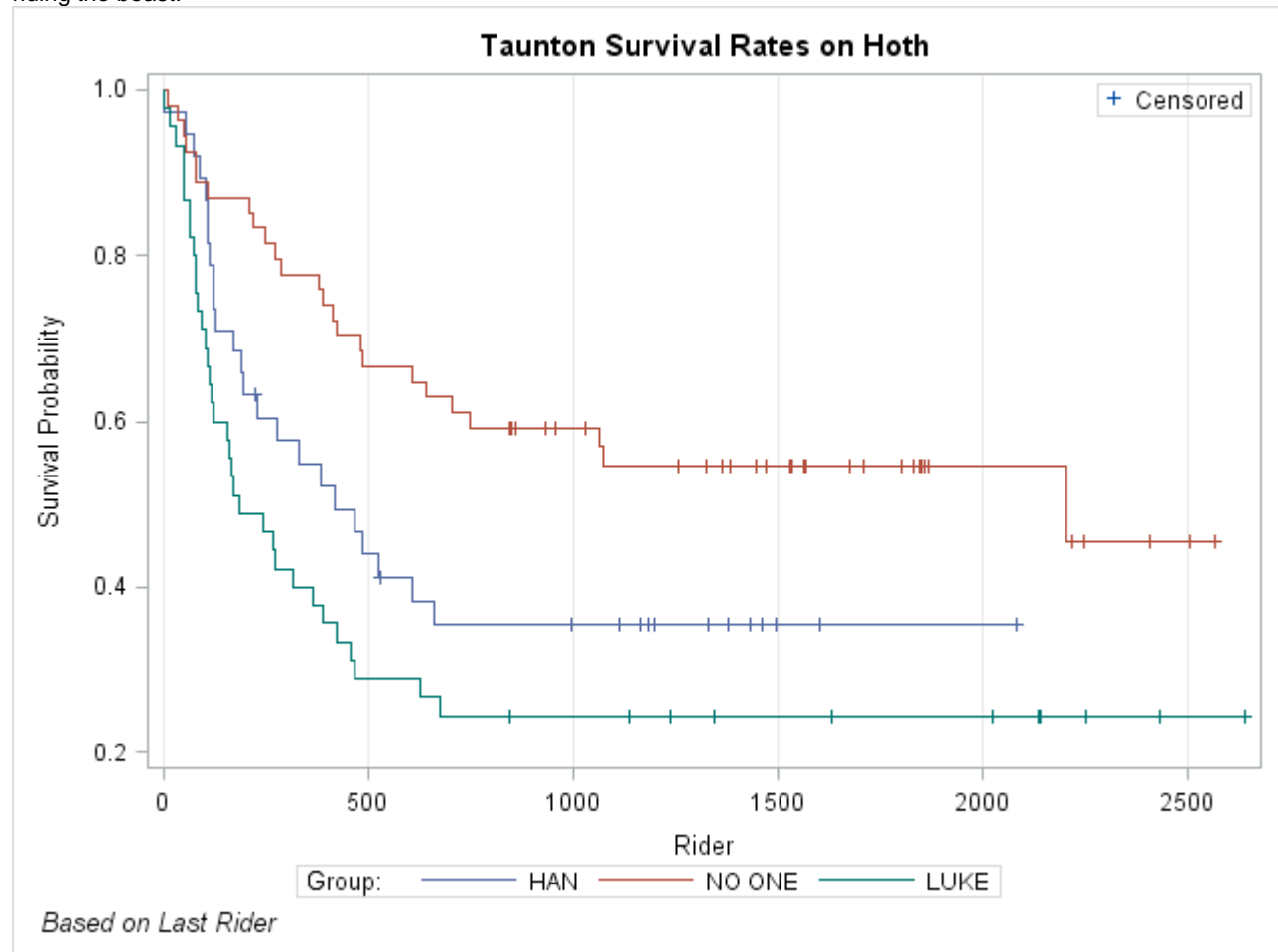


Figure 11. Survival Analysis for Tautons

I had to modify the WORK.BONEMARROW dataset to create the graph. Below is the code used to generate the graph after the changes were made.

```

proc template;
define statgraph sgdesign;
dynamic _T3 _CENSORED2 _T5 _CENSORED3 _T6 _SURVIVAL3 _GROUP5 _GROUP;
begingraph;
  entrytitle halign=center 'Taunton Survival Rates on Hoth';
  entryfootnote halign=left 'Based on Last Rider';
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10
columnngutter=10;
  layout overlay / xaxisopts=( griddisplay=on) yaxisopts=( label=('Survival
Probability'));
  stepplot x=_T6 y=_SURVIVAL3 / group=_GROUP5 name='step' connectorder=xaxis;
  scatterplot x=_T5 y=_CENSORED3 / name='scatter2' legendlabel='Censored'
markerattrs=(symbol=PLUS );
  scatterplot x=_T3 y=_CENSORED2 / group=_GROUP name='scatter'
markerattrs=(symbol=PLUS );
  discretelegend 'scatter2' / opaque=false border=true halign=right valign=top
displayclipped=true across=1 order=rowmajor location=inside;
  endlayout;
  sidebar / align=bottom spacefill=false;
  discretelegend 'step' / opaque=true border=true halign=center valign=center
title='Group:' displayclipped=true order=rowmajor;
  endsidebar;
  endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.BONEMARROW template=sgdesign;
dynamic _T3="T" _CENSORED2="CENSORED" _T5="T" _CENSORED3="CENSORED" _T6="T"
_SURVIVAL3="SURVIVAL" _GROUP5="GROUP" _GROUP="GROUP";
run;

```

PROC FALCON

One of the comedic and repeated sequences in the original three movies is Han Solo's horribly bad luck at jumping into hyperspace (also known as going to lightspeed). It seems that just when they need it, the engine has had some sort of malfunction; so, a Pareto chart graphically represents who is most responsible for the failure of the Falcon to "make the jump". Chewie is the nickname for Chewbacca, Han's copilot; R2D2 is a droid that has an ability to talk to the ship's computer, but occasionally does the wrong thing; Lando is the original owner of the Falcon and administrator for a mining operation that Han and the Rebels use as a refuge.

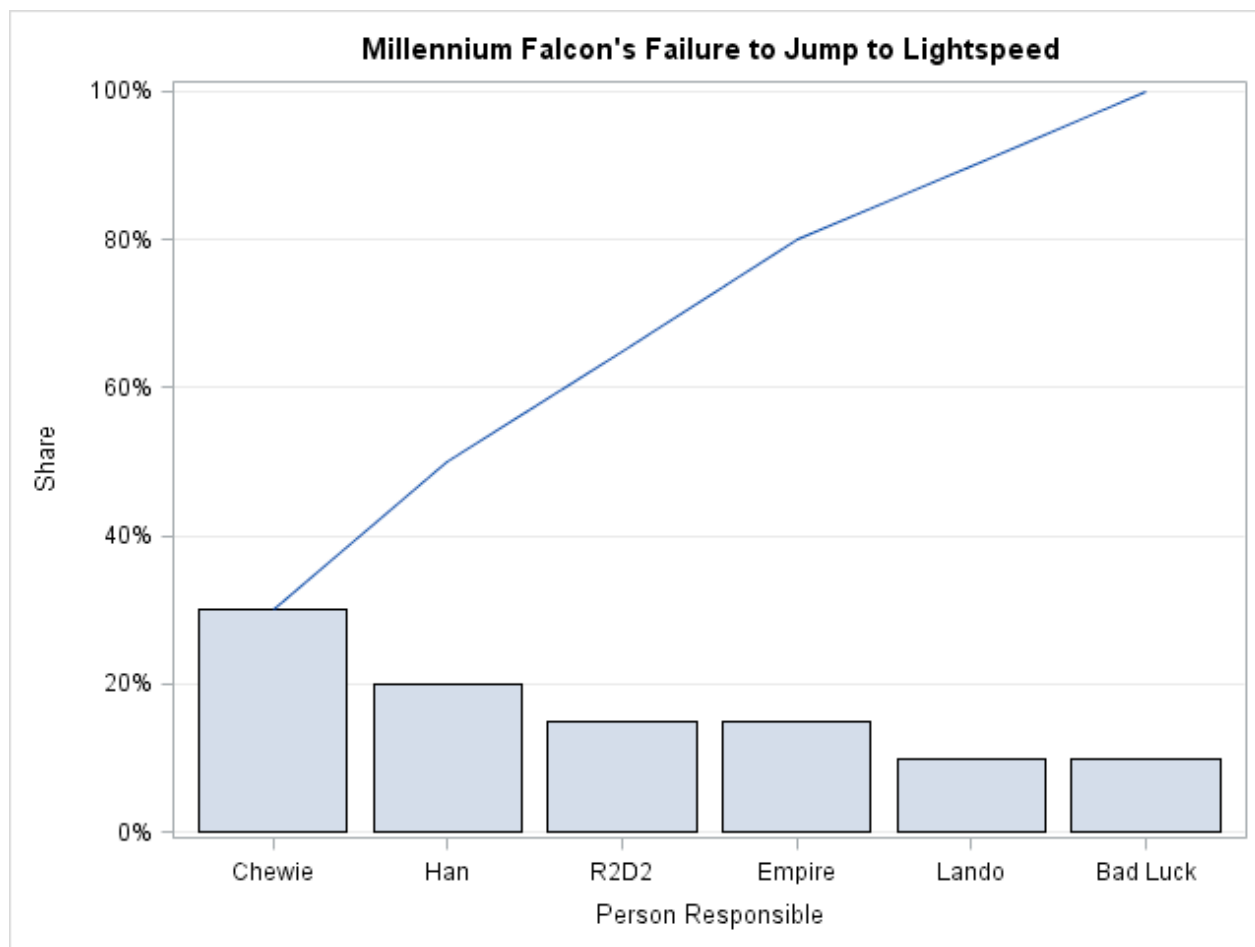


Figure 12. Pareto Chart of the Falcon's Hyperdrive Failures

Again I've made some minor changes to the pre-existing data set to fit my needs. In this case, it was the WORK.PARETO table.

```
proc template;
define statgraph sgdesign;
dynamic _PRODUCT _SHARE _PRODUCT2 _CUMULATIVE;
begingraph;
  entrytitle halign=center 'Millennium Falcon's Failure to Jump to Lightspeed';
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10
columnsgutter=10;
  layout overlay / yaxisopts=( griddisplay=on);
  barchart x=_PRODUCT y=_SHARE / name='bar' clusterwidth=1.0;
  seriesplot x=_PRODUCT2 y=_CUMULATIVE / name='series' connectorder=xaxis;
  endlayout;
endlayout;
endgraph;
end;
run;

proc sgrender data=WORK.PARETO template=sgdesign;
dynamic _PRODUCT="PRODUCT" _SHARE="SHARE" _PRODUCT2="PRODUCT"
_CUMULATIVE="CUMULATIVE" ;
run;
```

CONCLUSIONS

Star Wars is a fictional universe crossing all forms of media and 36 years; the franchise has millions of fans around the world, in all ages. SAS® is a powerful, multifunctional analysis software and two newer functions of SAS® are PROC TEMPLATE and Graphics Template Language and in learning these two features, I decided to combine these two seemingly unrelated areas of my life. Although very different from other PROCs in SAS, PROC TEMPLATE allows for enhanced flexibility within the SAS® Universe. It integrates very well with all other PROCs, allowing templates to be applied to tables, graphs, or any other output.

ACKNOWLEDGEMENTS

I'd like to thank the SAS® Canada Community for their inspiration, particularly Jared Prins, Barry Hong, Matt Malczewski and everyone else who's made a comment about Star Wars and analysis! A special thanks to Murphy Choy, who was brave enough as a section chair to allow me to present this!

REFERENCES

- Wikipedia, "Death Star", http://en.wikipedia.org/wiki/Death_star, February 1, 2013
- Haworth, L. SUGI31. PROC TEMPLATE: The Basics. Cary, NC: SAS Institute. Available at: <http://www2.sas.com/proceedings/sugi31/112-31.pdf>
- SAS Support, PROC TEMPLATE Tips Sheet. Available at <http://support.sas.com/rnd/base/ods/scratch/styles-tips.pdf> February 3, 2013
- Gupta, S. Using Styles and Templates to Customise SAS ODS Output. 2002. PharmaSUG Proceedings. Cary, NC: SAS Institute. Available at: <http://www.lexjansen.com/pharmasug/2002/proceed/tutorials/tu05.pdf>
- SAS Institute. SAS 9.2 Output Delivery System User's Guide. Cary, NC: SAS Institute. Available at: <http://support.sas.com/documentation/cdl/en/odsug/61723/PDF/default/odsug.pdf>