

# SCATTERPLOTS

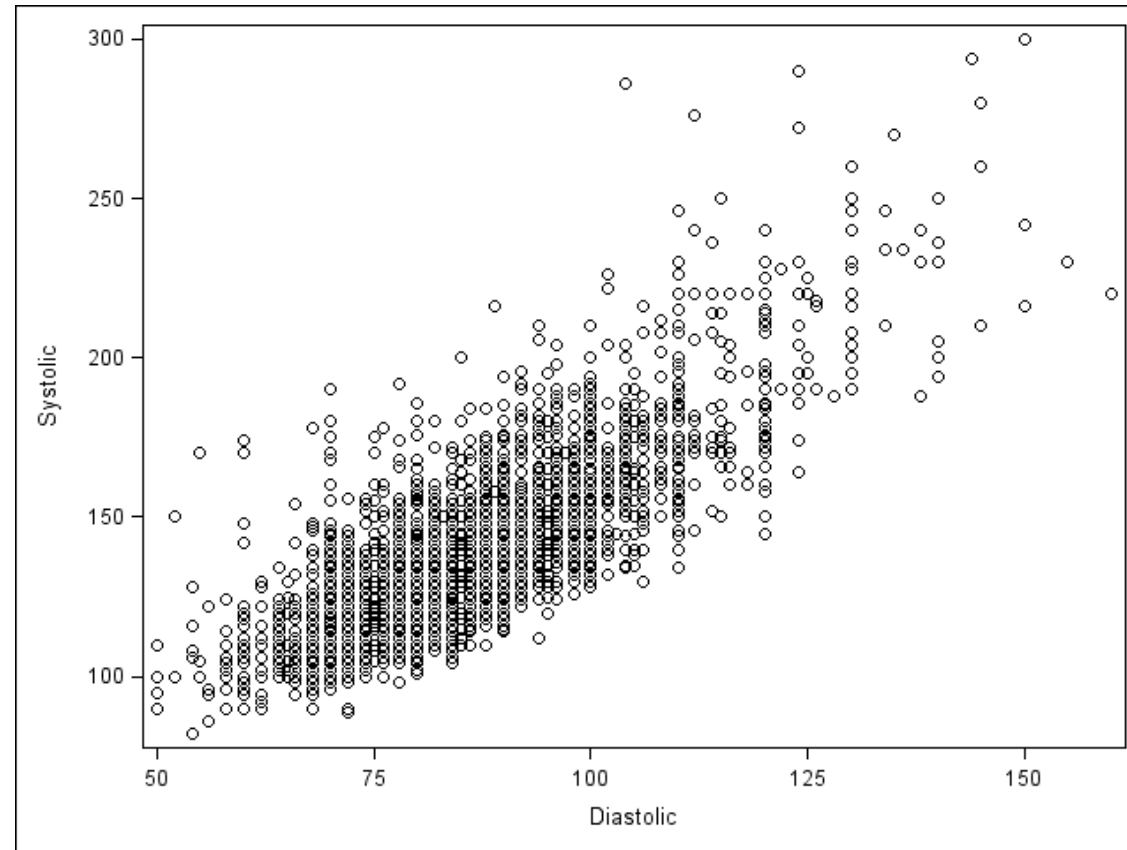
---

# Basic Scatterplot

- The SCATTER statement requires an x (horizontal) and y (vertical) variable specification:

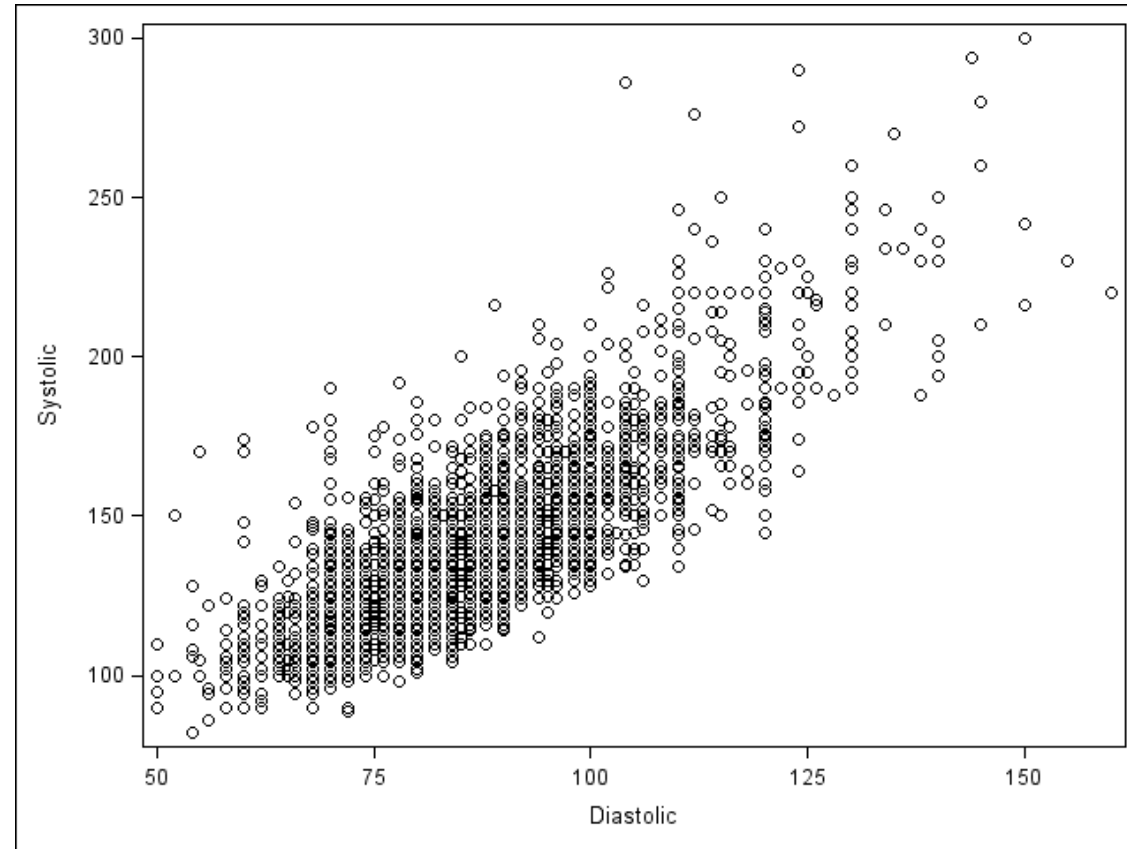
```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic;  
run;  
quit;
```

# Result



# Result

Default plotting symbol is an open circle, default color is black.



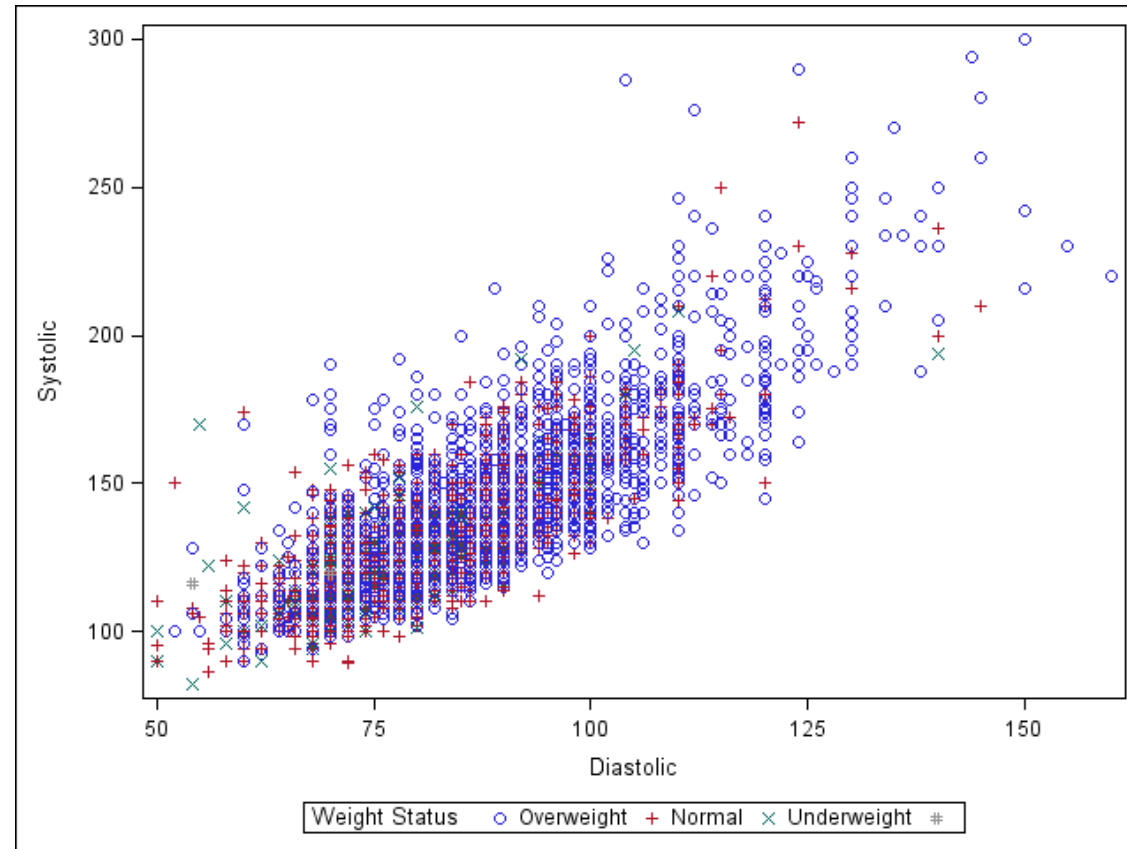
# Scatterplot Options

- Grouping is possible:

```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic / group=weight_status;  
run;  
quit;
```

# Result

Here both the color and symbol change across groups.

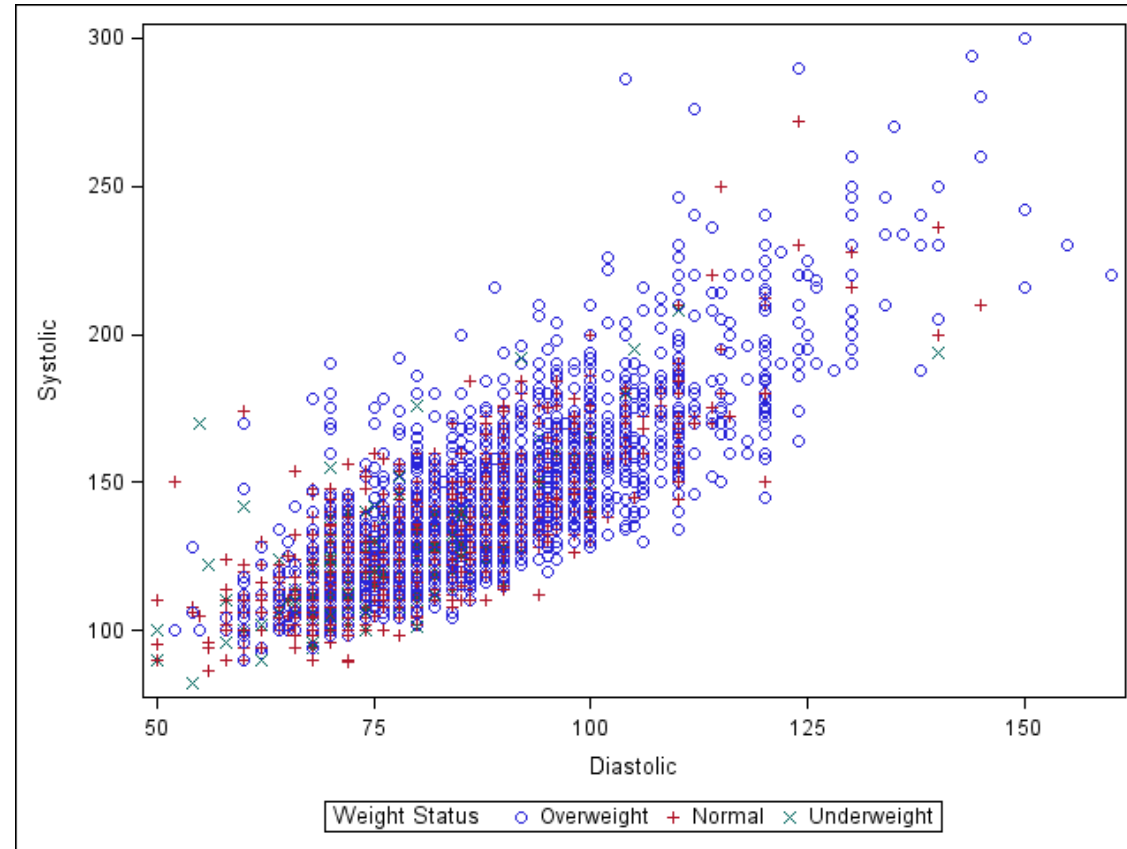


# Scatterplot Options

- This is a case where we might prefer the NOMISSINGGROUP option:

```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic / group=weight_status nomissinggroup;  
run;  
quit;
```

# Result



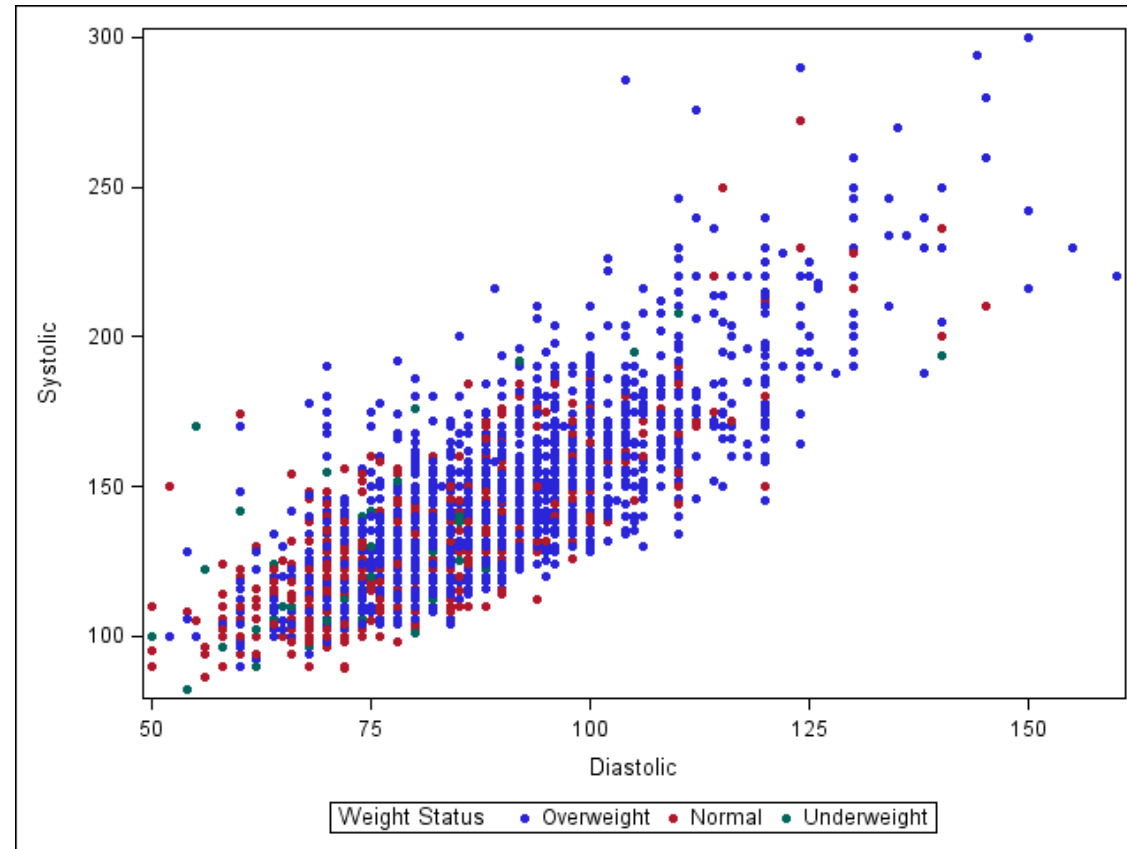


# Marker (Point) Options

- The markers placed down at each point have a variety of modifications that can be made to them, including an ATTRS option:

```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic / group=weight_status nomissinggroup  
           markerattrs=(symbol=circlefilled size=1.5mm) ;  
run;  
quit;
```

# Result

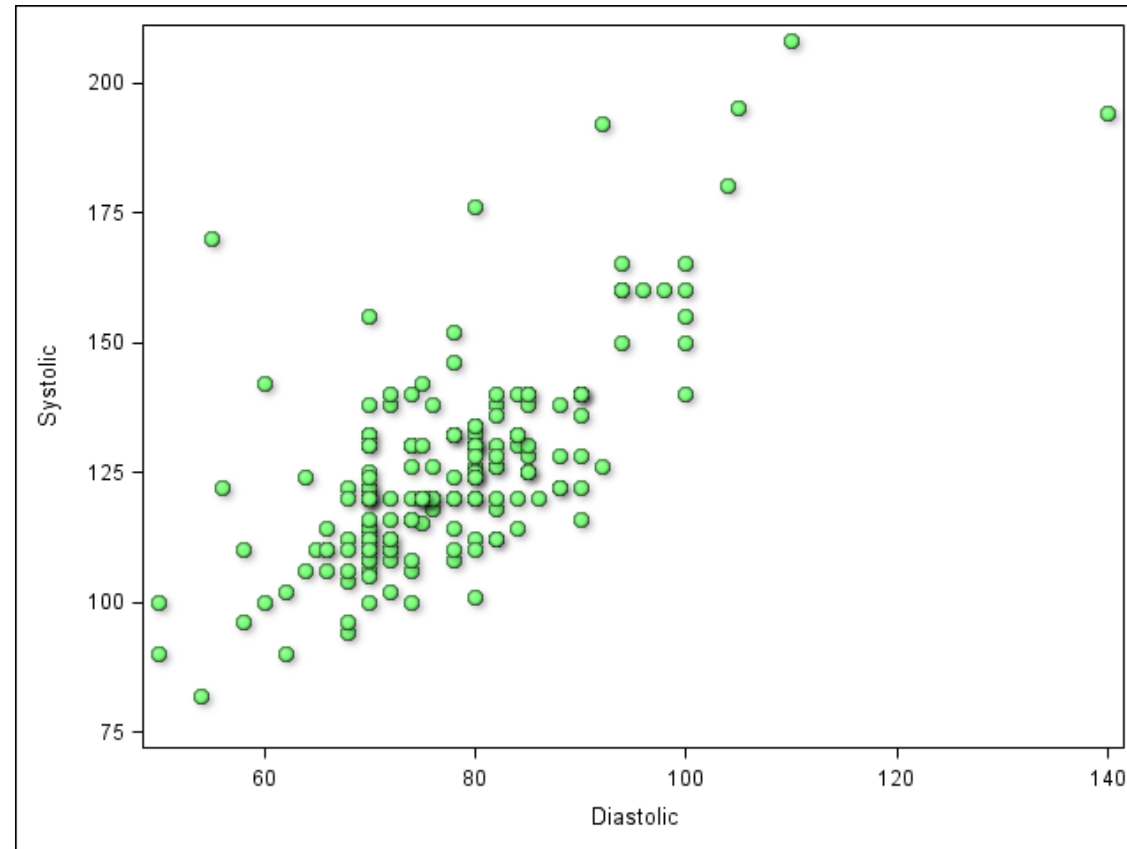


# Marker (Point) Options

- When using a filled symbol, it is possible to use DATASKIN:

```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic / markerattrs=(symbol=circlefilled  
                                                size=2.5mm color=cx66FF66) dataskin=sheen;  
    where weight_status='Underweight';  
run;  
quit;
```

# Result

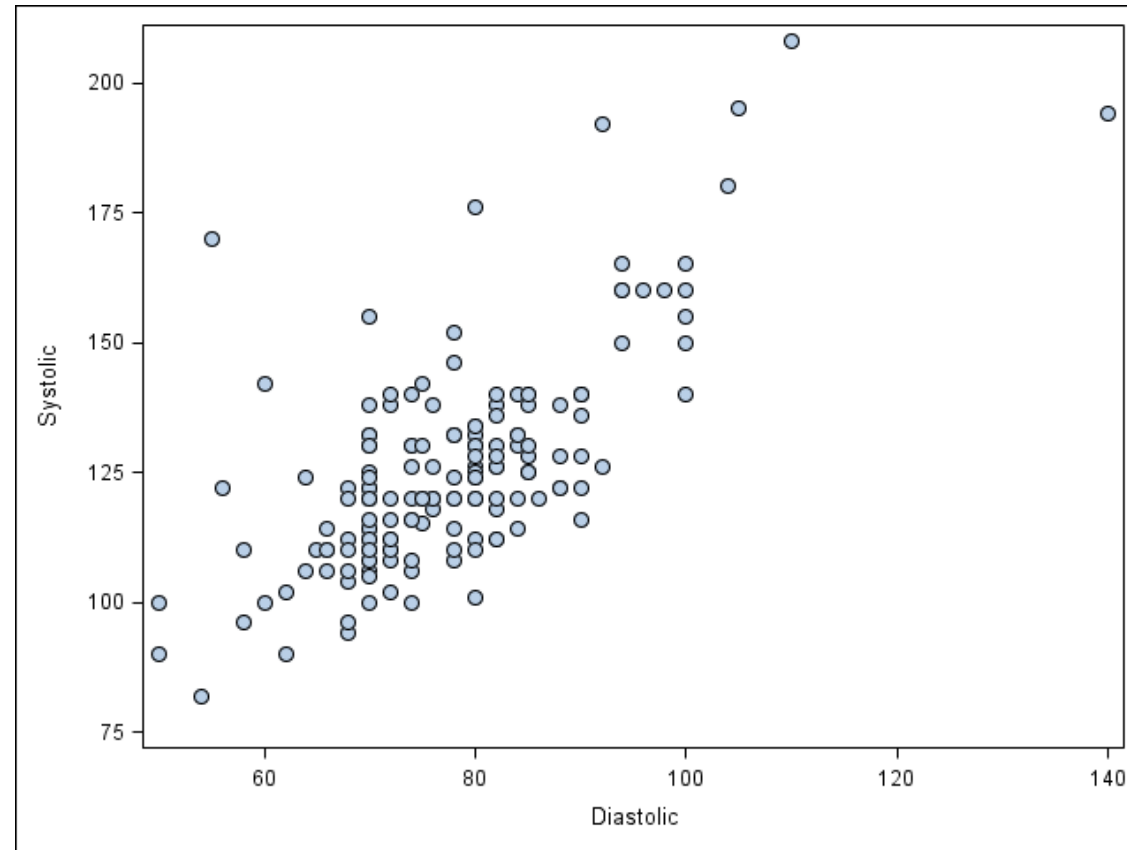


# Marker (Point) Options

- The FILLEDOUTLINEMARKERS option can also be used with filled symbols:

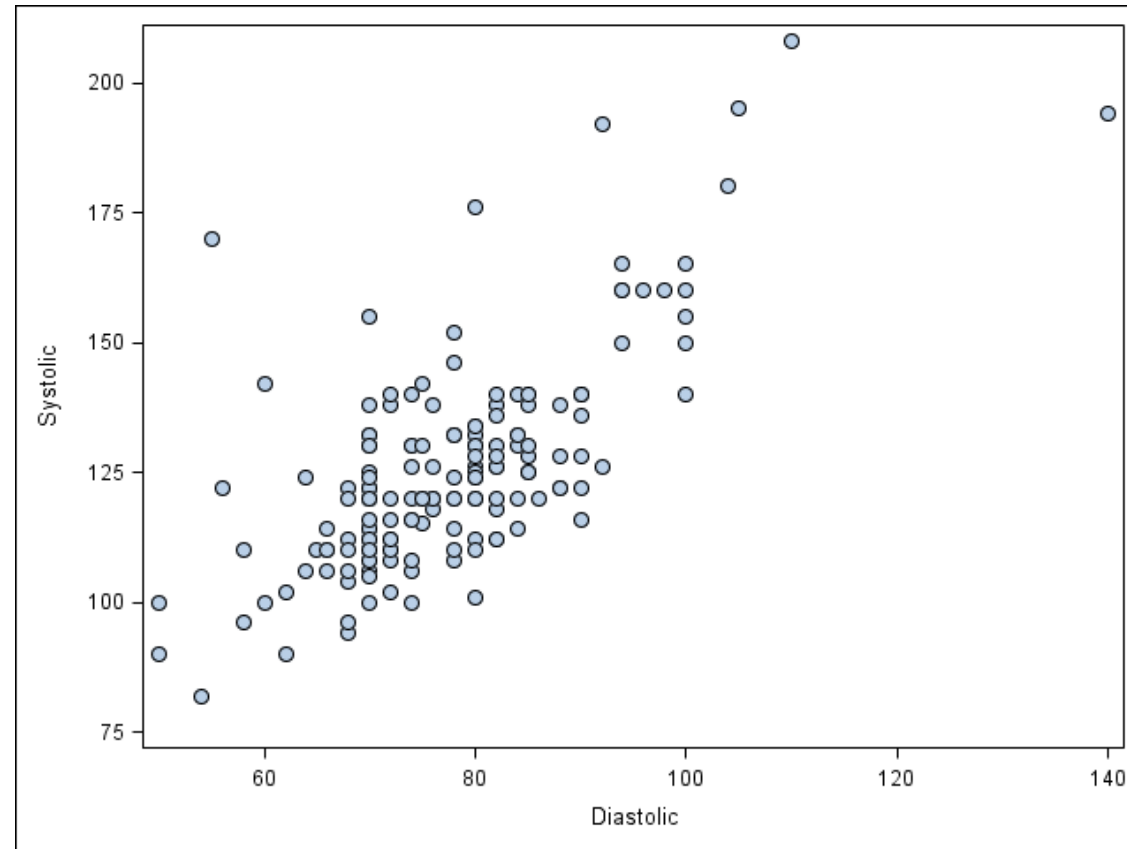
```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic / markerattrs=(symbol=circlefilled  
                                                size=2.5mm color=cx66FF66) filledoutlinedmarkers;  
    where weight_status='Underweight';  
run;  
quit;
```

# Result



# Result

Seems to ignore the color attribute.



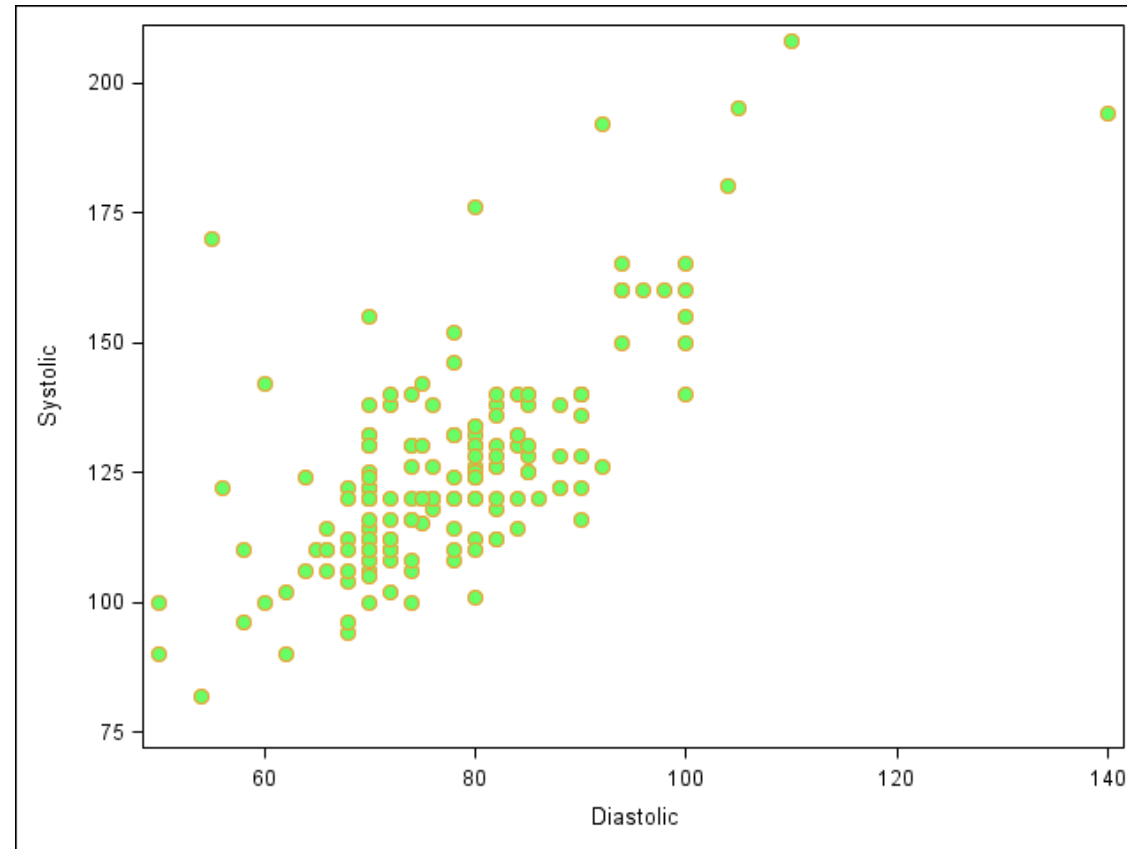
# Marker (Point) Options

- With FILLEDOUTLINEMARKERS the fill and outline attributes can be styled separately:

```
proc sgplot data=sashelp.heart;  
    scatter x=diastolic y=systolic / markerattrs=(symbol=circlefilled  
        size=2.5mm) filledoutlinedmarkers markerfillattrs=(color=cx66FF66)  
        markeroutlineattrs=(color=cxFF9933);  
    where weight_status='Underweight';  
run;  
quit;
```



# Result

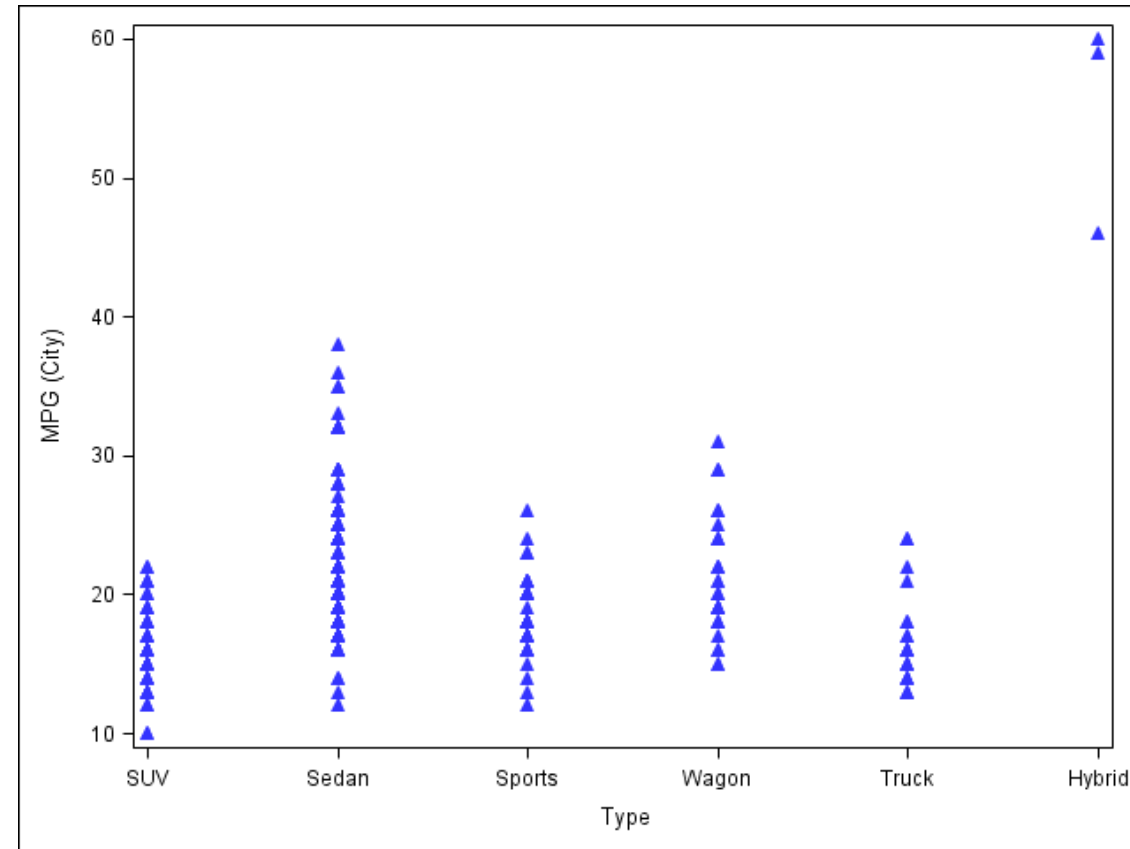


# More Scatterplots

- There is no requirement that each variable be quantitative:

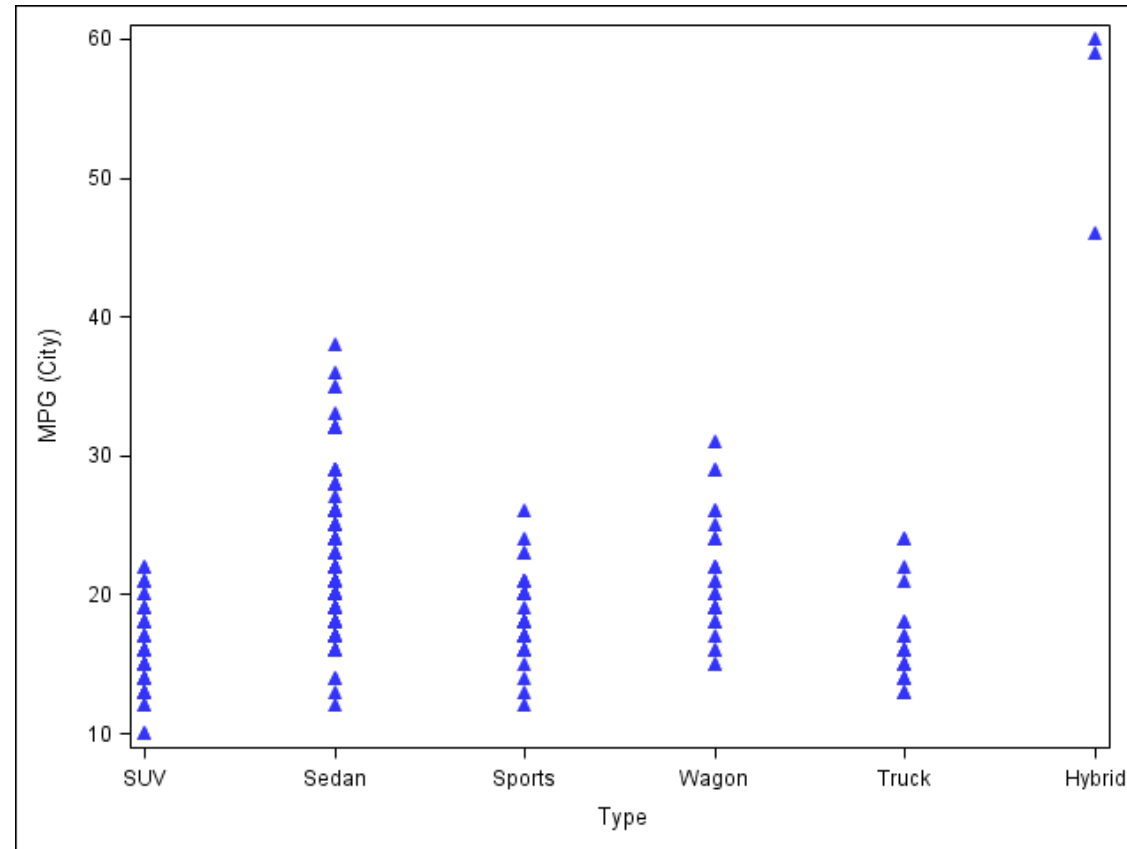
```
proc sgplot data=sashelp.cars;  
    scatter x=type y=mpg_city / markerattrs=(symbol=trianglefilled  
        color=cx3333FF) ;  
  
run;  
quit;
```

# Result



# Result

In this case, several observations occupy the same space.

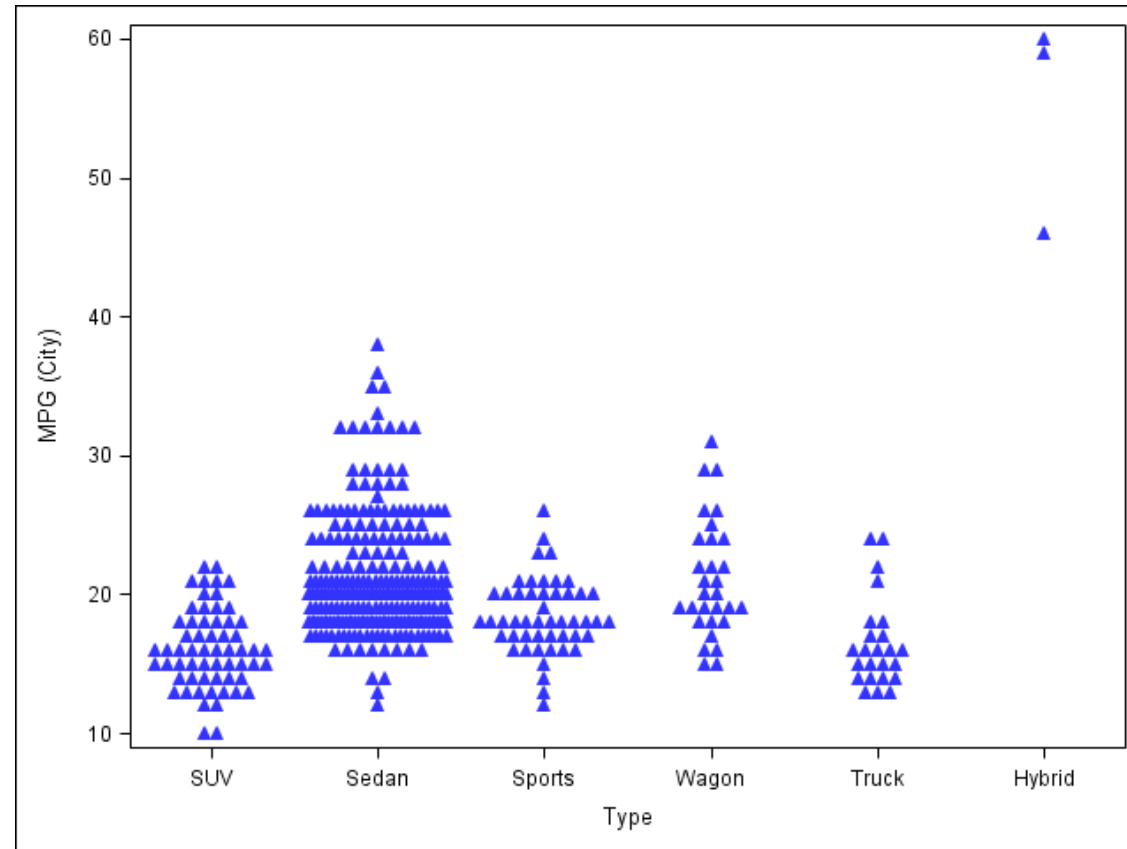


# More Options

- JITTER forces a separation between observations that land in the same place:

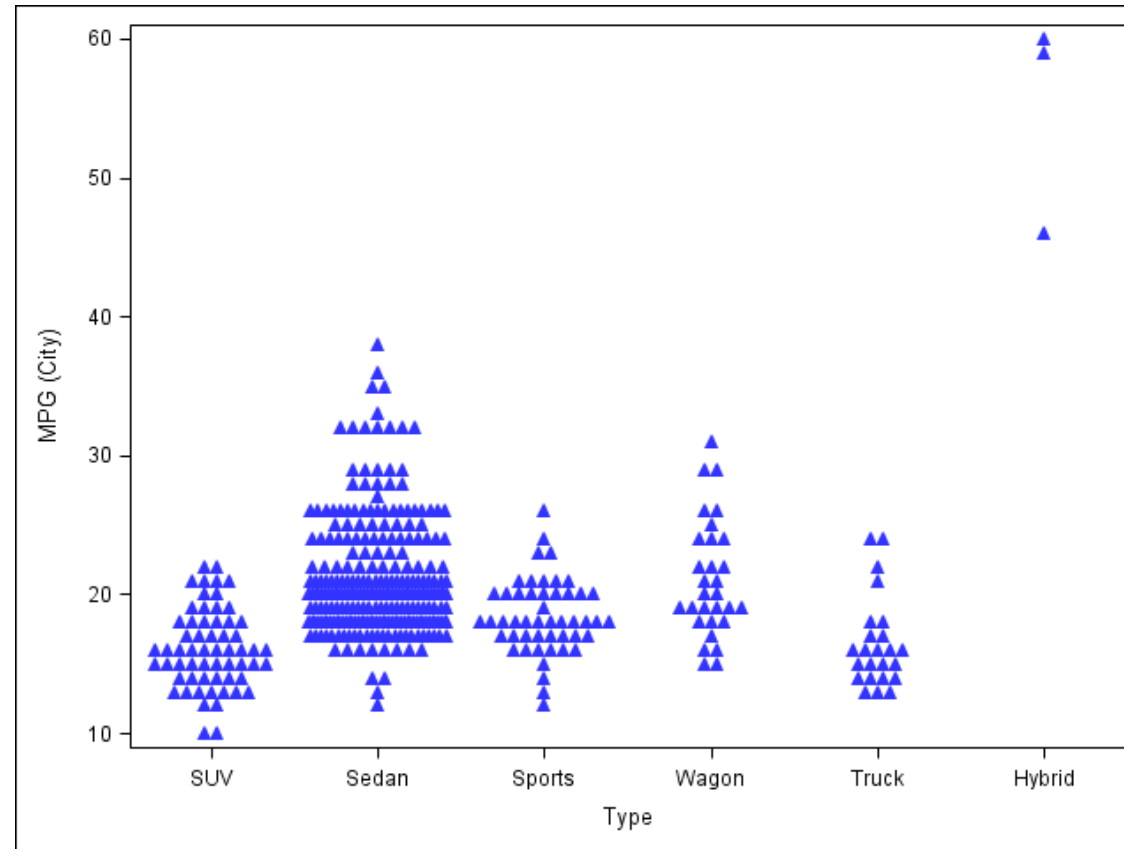
```
proc sgplot data=sashelp.cars;  
    scatter x=type y=mpg_city / markerattrs=(symbol=trianglefilled  
        color=cx3333FF) jitter;  
  
run;  
quit;
```

# Result



# Result

Jittering works in two dimensions as well, and the amount can be set by `JITTERWIDTH=`.



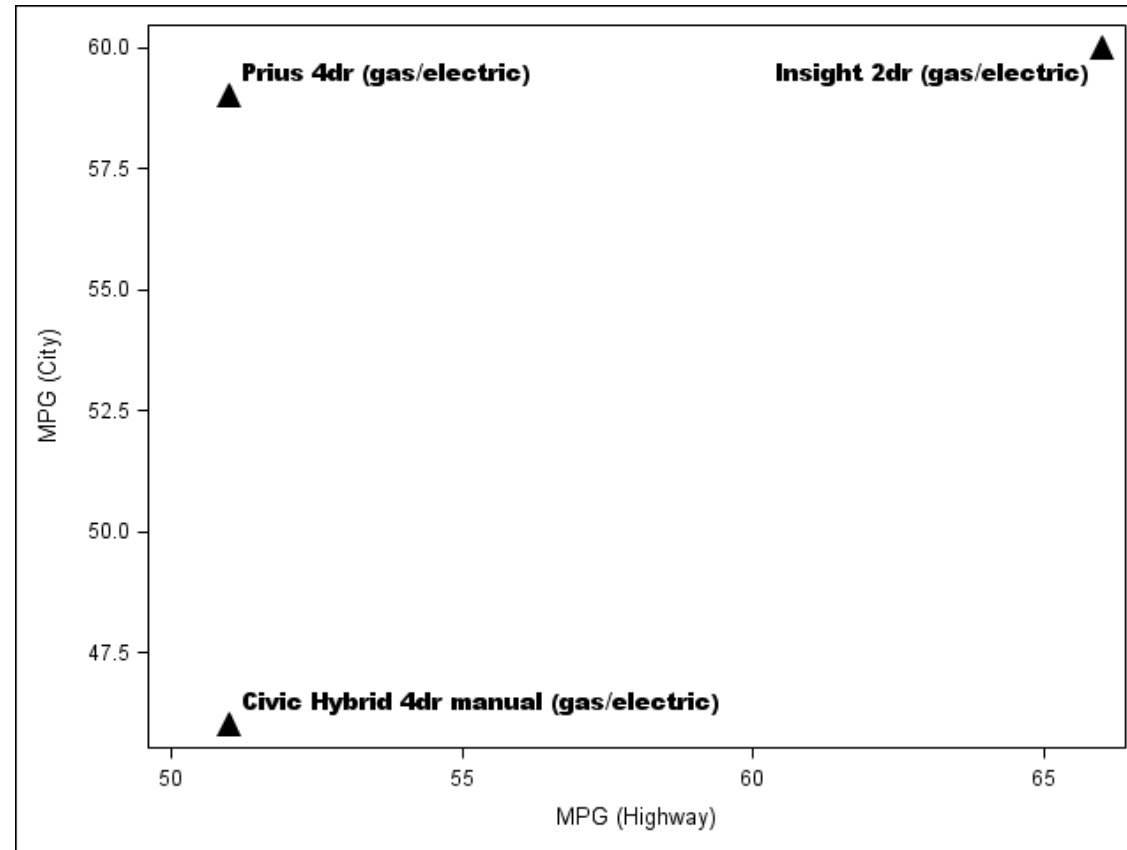
# More Options

- Labeling of data points is possible:

```
proc sgplot data=sashelp.cars;  
    scatter x=mpg_highway y=mpg_city/markerattrs=(symbol=trianglefilled size=10pt)  
            datalabel=model datalabelattrs=(family='Arial Black' size=10pt);  
    where type='Hybrid';  
run;  
quit;
```

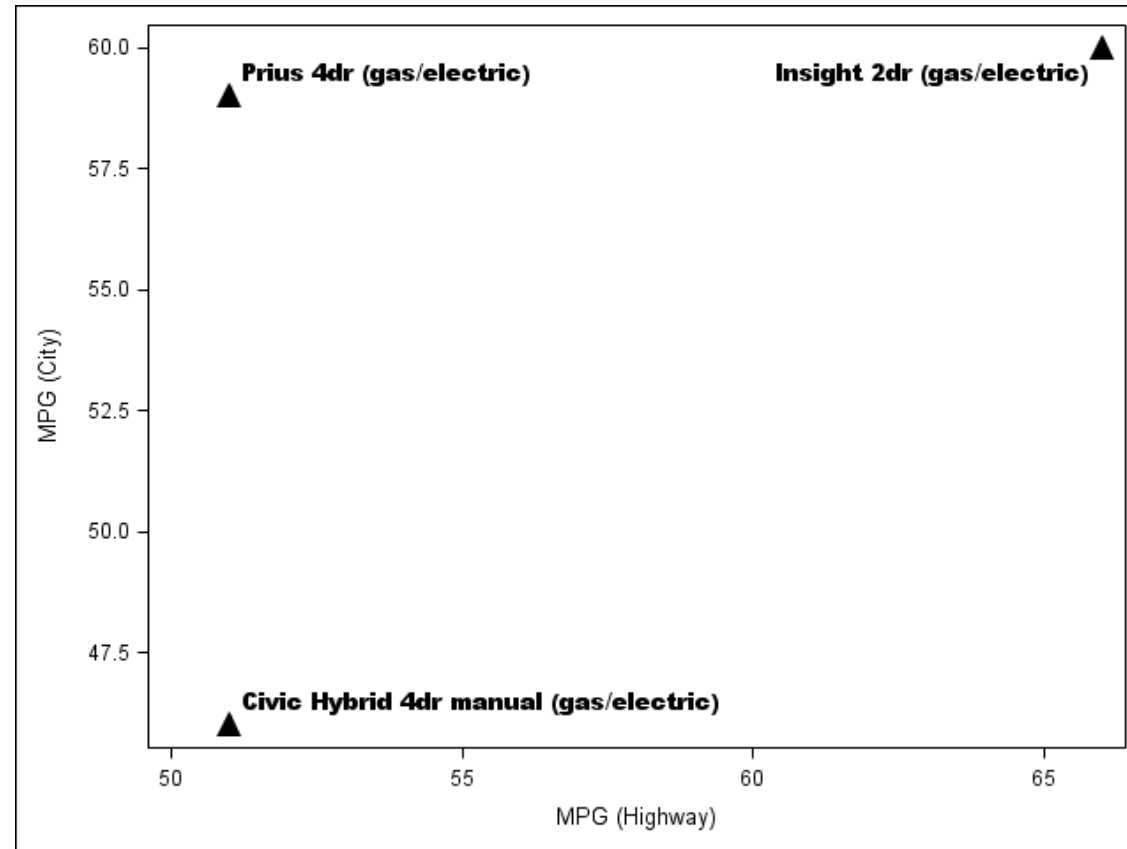


# Result



# Result

If no variable is included (DATALABEL),  
the vertical coordinate is used

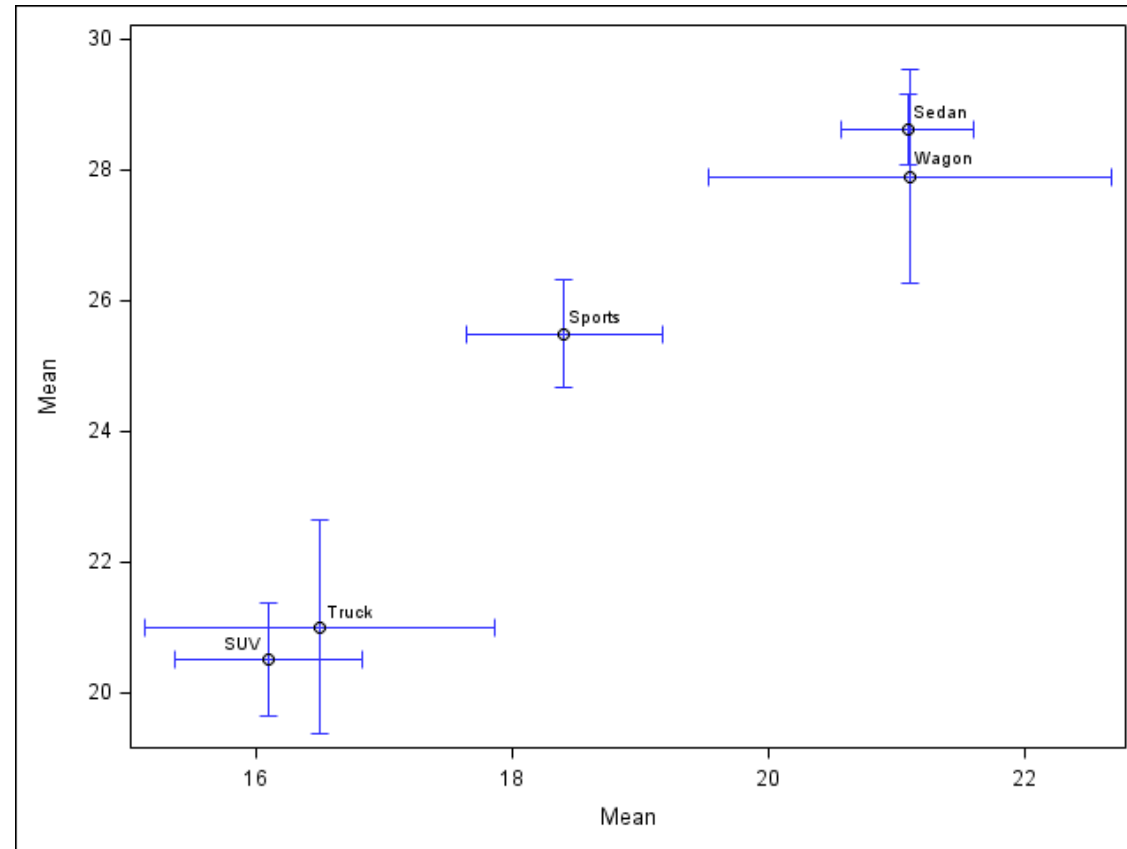


# More Options

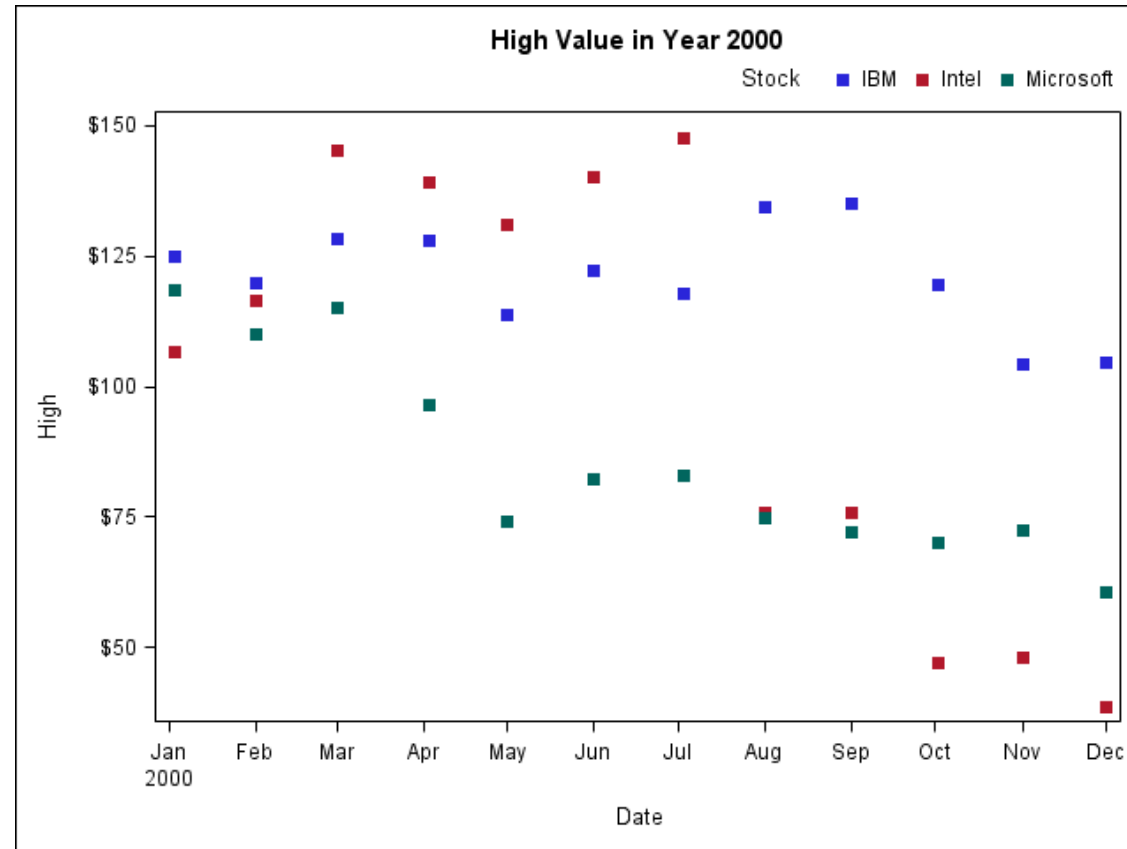
- Error bars are possible, too (in both directions). The values must be pre-computed and on the input data set:

```
ods listing close;
proc means data=sashelp.cars lclm mean uclm;
  where type ne 'Hybrid';
  class type;
  var mpg_city mpg_highway;
  ods output summary=results;
run;
ods listing;
proc sgplot data=results;
  scatter x=mpg_city_mean y=mpg_highway_mean / xerrorlower=mpg_city_lclm
    xerrorupper=mpg_city_uclm yerrorlower=mpg_highway_lclm
    yerrorupper=mpg_highway_uclm errorbarattrs=(color=cx3333FF)
    datalabel=type;
run;quit;
```

# Result



# Exercise 1



# Exercise 2

