

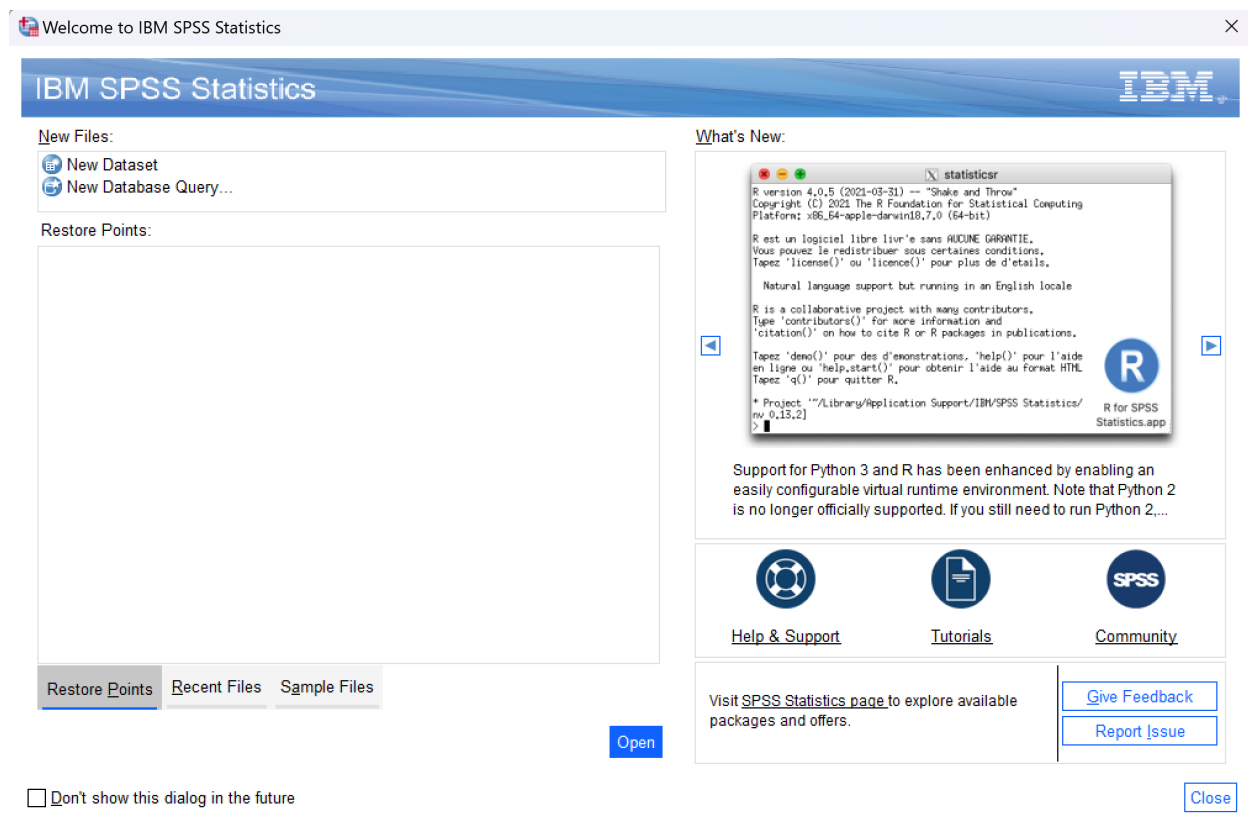
Instructions for IBM® SPSS® Statistics

IBM SPSS Statistics, or “SPSS” for short, is produced by IBM, Inc. and can be obtained [here](#).

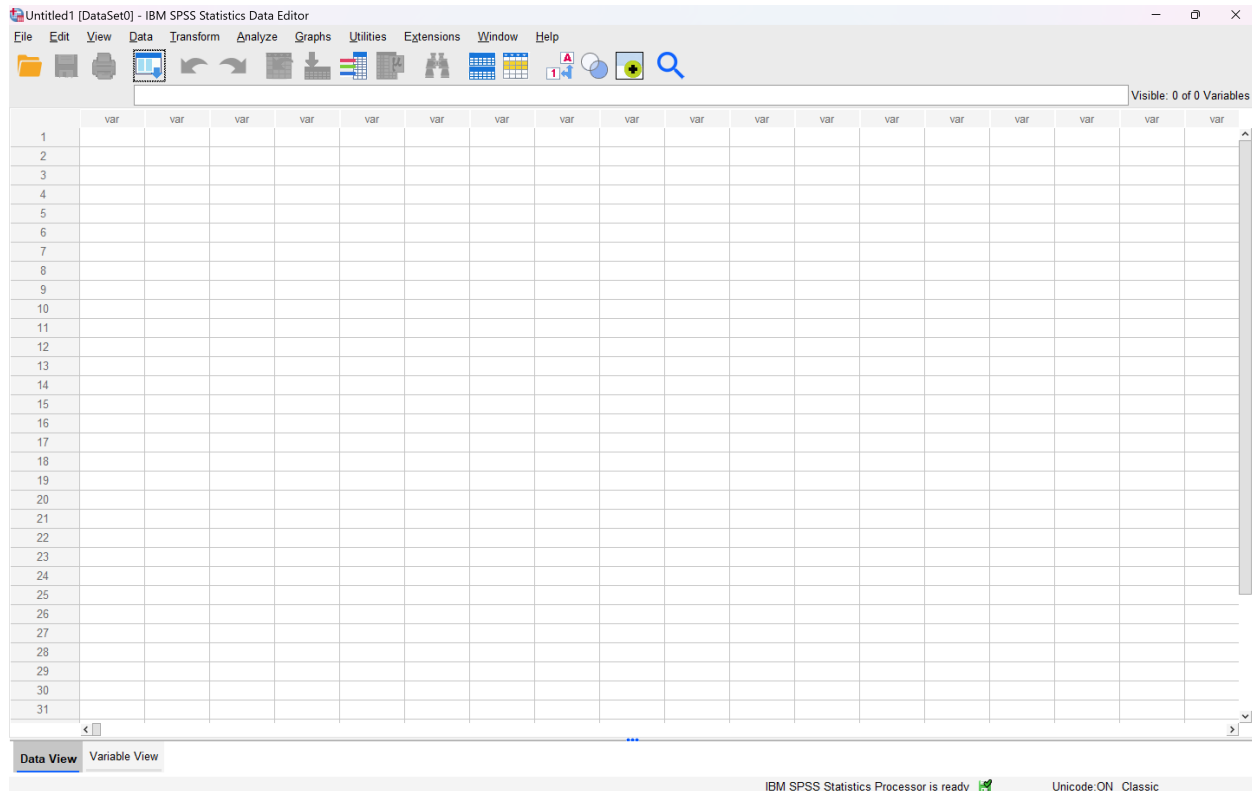
1. Appearance

[Note: The screen shots provided below are from Windows version 28 of SPSS. Versions for other operating systems (e.g. for MacOS) may appear slightly different.]

Once downloaded and SPSS is launched you may see an opening dialog box like that shown below:



Close this dialog box and you will see the “data window” screen, shown below, which looks like a simple spreadsheet:



This is one of two primary windows SPSS uses. The second is the “output window” which shows what SPSS has done for you after issuing commands, (we will see an example of this this screen shortly).

Notice at the bottom there are two tabs, “Data View” and “Variable View”. Data View shows the data that are currently in memory (if any), while Variable View will show characteristics about each variable (e.g., whether the variable is a numeric or string, number of missing values, etc.).

Note that data in SPSS is typically organized such that each column contains data for a particular variable, and each row represents an observation’s values for variables. The first column, shown as “1”, “2”, “3”,... are observation numbers.

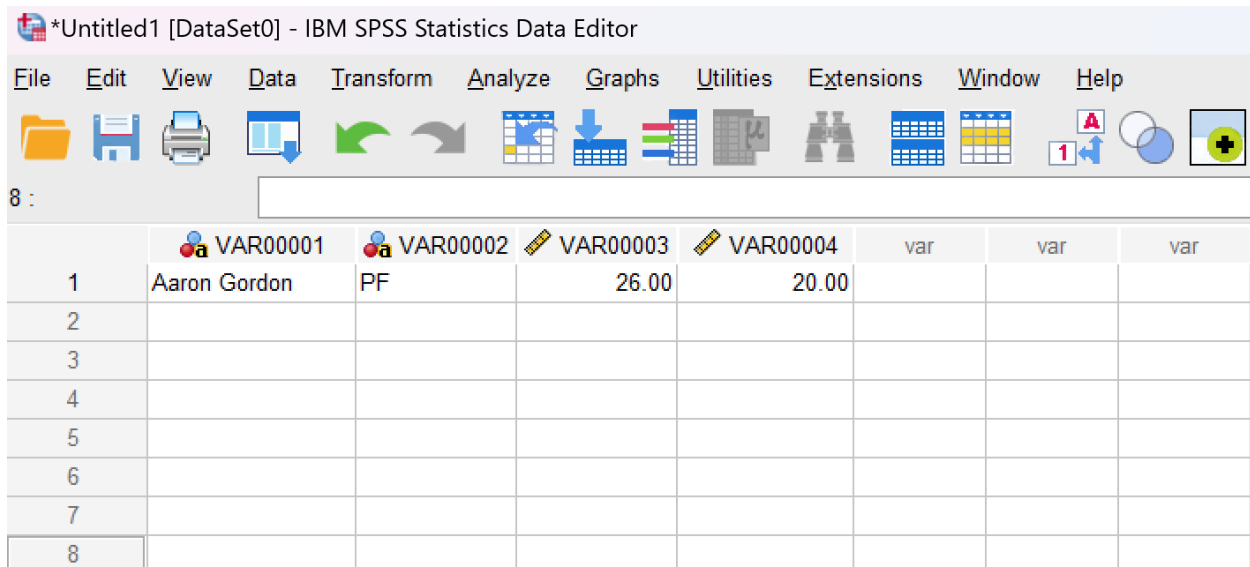
2. Entering Data

Manually

Two primary ways to enter data into SPSS are to type the data manually, or to read a data file into SPSS. As an example, we can enter data from our NBA data set (“nba2021_22.csv”, available on the *Regression Basics* companion website) manually. Here is some of the information about our first player in the data set:

player:	Aaron Gordon
position:	PF
age:	26
salary:	20

We can begin by just typing each piece of information into the Data View spreadsheet in SPSS:



We could do the same for all the other variables for this player. Notice that the variable names are generated generically (e.g., VAR00001). These can be changed by going to the Variable View tab and replacing the generic names with more useful ones:

*Untitled1 [DataSet0] - IBM SPSS Statistics Data Editor

File Edit View Data Transform Analyze Graphs Utilities Extensions Window Help

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	player	String	12	0		None	None	12	Left	Nominal	Input
2	position	String	2	0		None	None	9	Left	Nominal	Input
3	age	Numeric	8	2		None	None	9	Right	Unknown	Input
4	VAR00004	Numeric	8	2		None	None	10	Right	Unknown	Input
5											
6											

Notice that as the data were typed into SPSS, the program assigned the data a “Type” (i.e., “String” or “Numeric”) based on what was entered. Returning to the Data View window we see that the generic names have now been replaced:

*Untitled1 [DataSet0] - IBM SPSS Statistics Data Editor

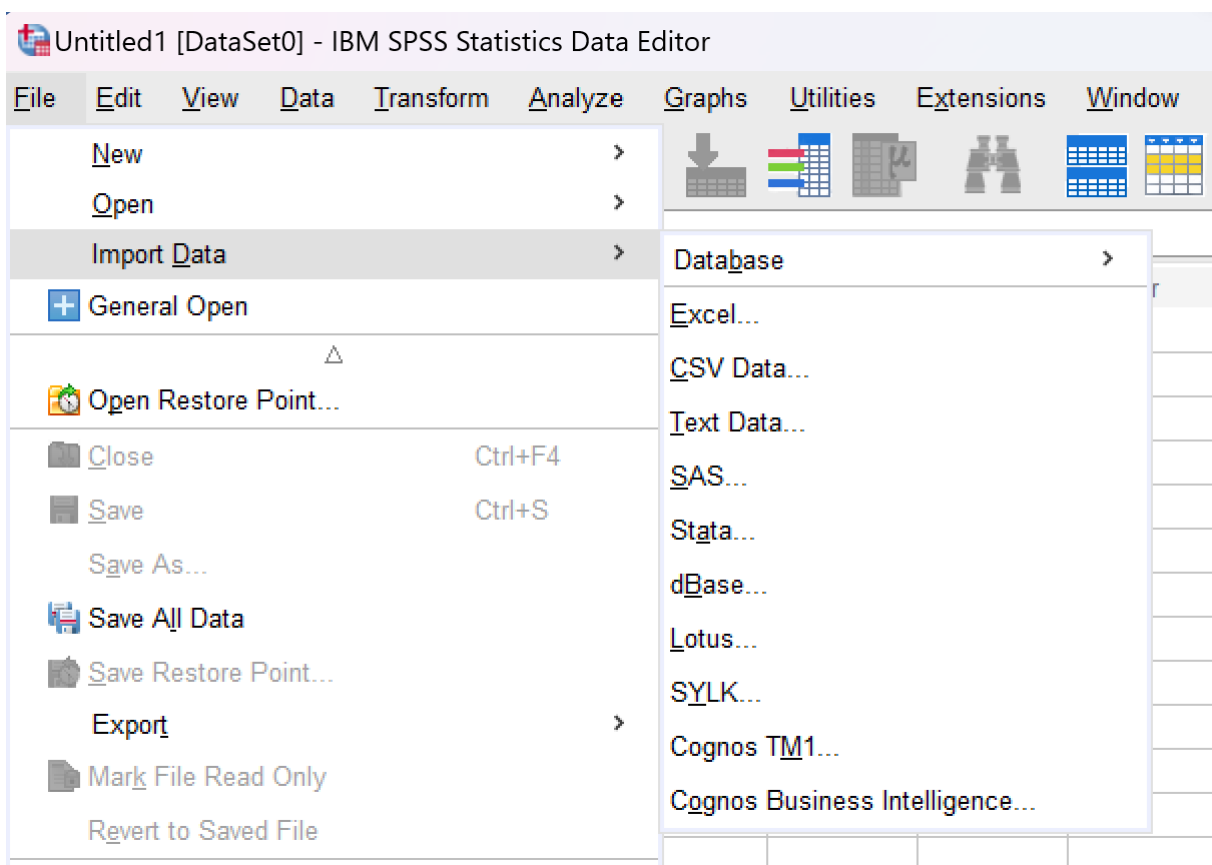
File Edit View Data Transform Analyze Graphs Utilities Extensions Window Help

	player	position	age	VAR00004	var	var	var
1	Aaron Gordon	PF	26.00	20.00			
2							
3							
4							
5							
6							
7							
8							


We would continue to enter all values for the first player (Aaron Gordon), then move to the second row to enter the data for the next player (Aaron Holiday) in a similar way, and so on.

Reading Data Files Into SPSS

While entering data into SPSS manually may be necessary in some cases, most often we will be working with data sets that can be read into SPSS as a whole. SPSS is capable of importing a variety of file types, including Excel, CSV, Text, SAS, Stata, and others. Importing data into SPSS can be done using the drop-down menus. Begin by clicking File while in the Data View window, then click Import Data, and then choose a file type:



Continuing with our NBA data, given it is a .csv file type, we would choose CSV Data. This will open a dialog box where we can navigate to the folder where the data file is stored, and then select the file named “nba2021_22.csv”. This will bring up a preview window that gives you a view of what is being read into SPSS:

 Read CSV File
 ✕

File: nba2021_22.csv

```

player,position,age,salary,team,weight,height,games,minspg,orebsp,grebsp,astspg,slspg
Aaron Gordon,PF,26,20,ORL,235,80,453,28.0125,1.5625,4.625,6.1875,2.5375,0.725,0.65,1.5125,1.9
Aaron Holiday,PG,25,2.61934,IND,185,72,182,18.4,0.2,1.43333,1.66667,2.33333,0.633333,0.233333
Aaron Nesmith,SF,22,4.13205,BOS,215,77,46,14.5,0.6,2.2,2.8,0.5,0.3,0.2,0.5,1.9,4.7,1
Al Horford,C,35,27.25,ATL,240,81,881,32.1786,2.03571,6.01429,8.05,3.24286,0.828571,1.18571,1.5
Alec Burks,SG,30,10.0128,UTH,214,78,544,21.6923,0.553846,2.73077,3.27692,1.84615,0.623077,0.
Aleksiej Pokusevski,PF,20,3.58728,OKC,190,84,45,24.2,0.7,4.4,7.2,2.2,0.4,0.9,2.2,1.3,8.2,1
    
```

☒ First line contains variable names

☐ Remove leading spaces from string values

☐ Remove trailing spaces from string values

 Delimiter between values: Comma

 Decimal symbol: Period

 Text Qualifier: Double quote

 Percentage of values that determine data type: 95

☒ Cache data locally

Advanced Options(Text Wizard)

OK

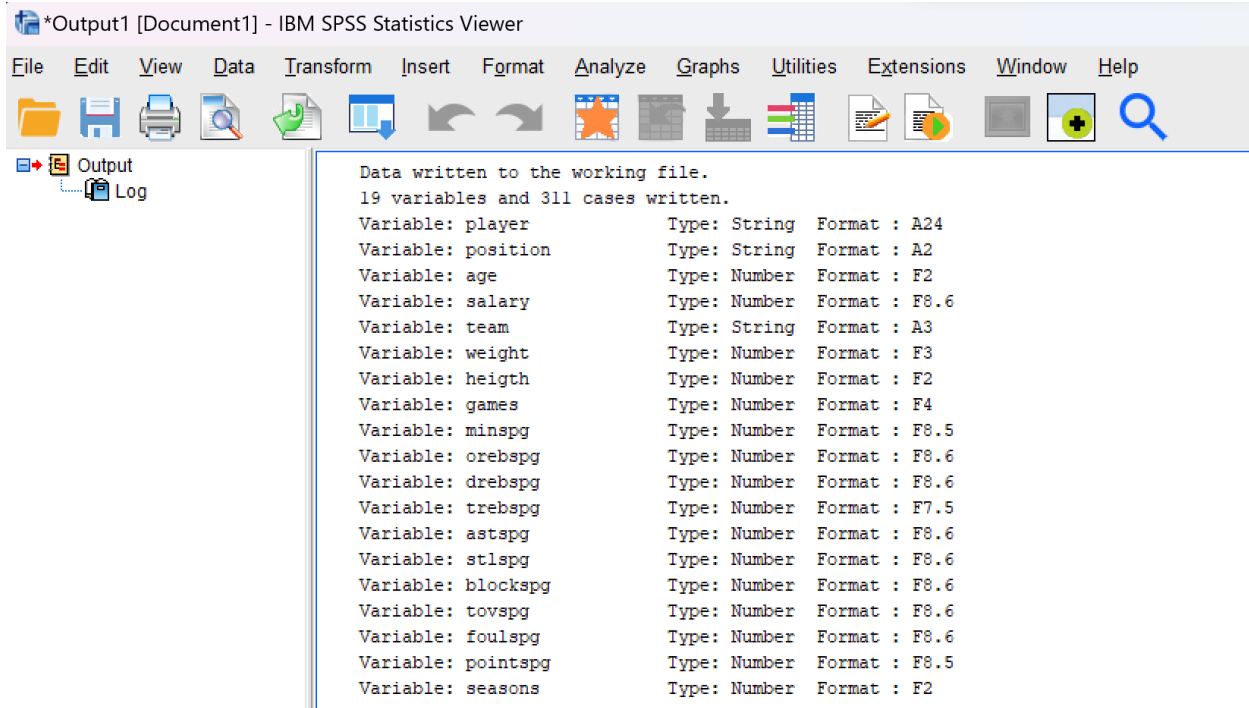
Paste

Reset

Cancel

Help

Click OK and the data will be read into SPSS's Data View window. This step normally brings up the "output window" which shows the user what SPSS has done with this command. A portion of this window is shown below:



Minimizing this window will bring the Data View window to the forefront. We can now see that SPSS has read all the columns and rows of the “nba2021_22.csv” file (a portion is shown below):

	player	position	age	salary	team	weight	height	games	minspg	orebsp	drebsp	trebsp	astspg
1	Aaron Gordon	PF	26	20.000000	ORL	235	80	453	28.01250	1.562500	4.625000	6.18750	2.537500
2	Aaron Holiday	PG	25	2.619340	IND	185	72	182	18.40000	.200000	1.433330	1.66667	2.333330
3	Aaron Nesmith	SF	22	4.132050	BOS	215	77	46	14.50000	.600000	2.200000	2.80000	.500000
4	Al Horford	C	35	27.250000	ATL	240	81	881	32.17860	2.035710	6.014290	8.05000	3.242860
5	Alec Burks	SG	30	10.012800	UTH	214	78	544	21.69230	.553846	2.730770	3.27692	1.846150
6	Aleksej Pokusevski	PF	20	3.587280	OKC	190	84	45	24.20000	.700000	4.000000	4.70000	2.200000
7	Alex Caruso	SG	27	9.245000	LAL	186	77	184	18.95000	.500000	1.850000	2.32500	2.450000
8	Alex Len	C	28	3.825300	PHX	250	84	531	17.48000	1.710000	3.690000	5.41000	.750000
9	Andre Iguodala	SF	38	2.641690	GSW	215	78	1192	31.65290	.894118	3.976470	4.87059	4.123530
10	Andrew Wiggins	SF	26	29.542000	MIN	205	79	525	35.13750	1.250000	3.250000	4.51250	2.575000
11	Anfernee Simons	SG	22	2.543980	POR	181	75	154	15.03330	.266667	1.433330	1.70000	1.166670
12	Anthony Edwards	SG	20	11.067800	MIN	225	77	72	32.10000	.800000	3.800000	4.70000	2.900000
13	Anthony Gill	PF	29	1.208150	WAS	230	80	26	8.40000	.600000	1.300000	2.00000	.400000
14	Austin Rivers	PG	29	2.401540	LAC	200	76	588	24.24170	.316667	1.808330	2.12500	2.291670
15	Avery Bradley	PG	31	2.641690	BOS	180	75	598	26.67860	.550000	2.228570	2.76429	1.850000

In addition, the Variable View tab will now show all the variables that were read into SPSS with, among other things, their variable type (e.g., “String” or “Numeric”). At this point the user would normally save the data set as an SPSS file (.sav) by clicking File, and then Save As... This will

bring up a dialog box where you can give the file a name (e.g., “nba.sav”), and choose the location where you would like the SPSS data file to be saved. Later the file can be opened by starting SPSS, clicking File, then Open, then Data..., and then navigating to the destination where the file is stored. Alternatively, the user can simply navigate to the saved SPSS data file (e.g., “nba.sav”) and then double click it. This will launch SPSS and the file will open to the Data View window with the data in memory.

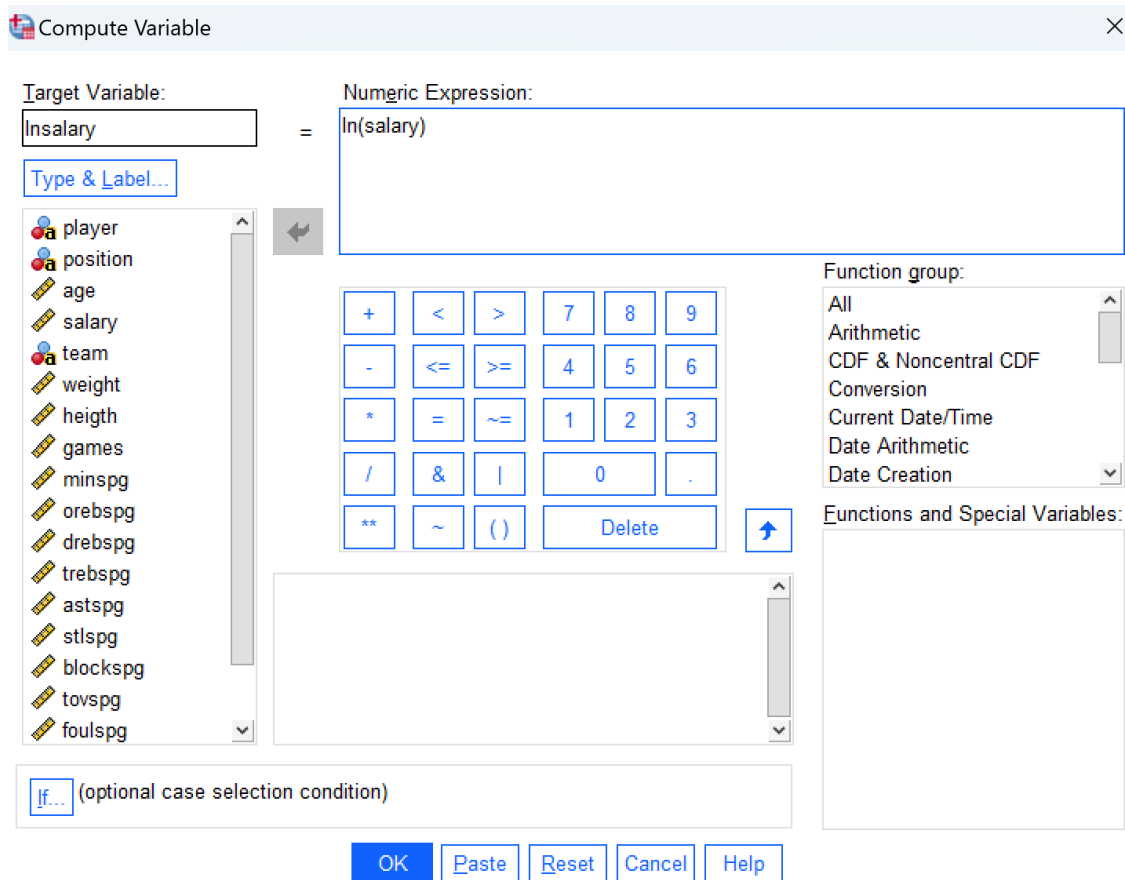
3. Transforming Variables

Once we have entered data into SPSS, we may want to transform some variables. For example, we may want to compute the natural log of a variable or compute the squared value of a variable. To do so we begin on the Data View page, then choose Transform on the main menu, then choose Compute Variable..., this will bring up a dialog box that can be used to transform an existing variable. Working with our NBA data, we would have the dialog box shown below:

The screenshot shows the IBM SPSS Statistics Data Editor with a dataset named 'nba.sav'. The 'Compute Variable' dialog box is open, allowing the user to create a new variable. The 'Target Variable' box is empty, and the 'Numeric Expression' box is also empty. The 'Type & Label' list on the left contains variables from the dataset, including 'player', 'position', 'age', 'salary', 'team', 'weight', 'height', 'games', 'minspg', 'orebsp', 'drebsp', and 'trebsp'. The 'Function group' dropdown is set to 'All', and the 'Functions and Special Variables' list on the right shows various mathematical and statistical functions. The data view in the background shows columns for player, position, age, salary, team, weight, height, games, minspg, orebsp, drebsp, and trebsp, with rows for 28 players.

The “Target Variable” is the variable that we would like to create. Suppose we wish to compute the natural log of salary and call it *lnsalary*. We would begin by typing *lnsalary* into the Target Variable box. In the “Numeric Expression:” box we would type $\ln(\text{salary})$ and then hit OK.

[Alternatively, we could type: $\ln($ and then highlight the variable salary and use the arrow key to insert this variable into the expression, then add the closing parentheses, $)$ and hit OK. This method may be easier when the computation of the new variable is complicated and involves multiple existing variables.]

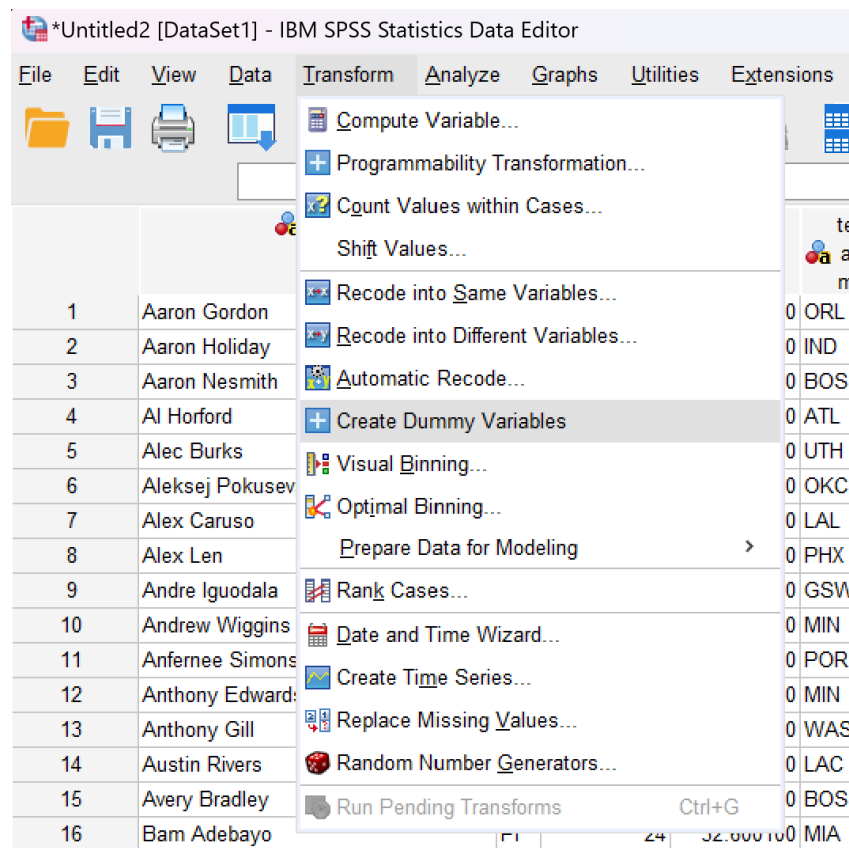


A new column of data has now been created, named `lnsalary`, that now contains the natural log of the salary values. Other transformations can be carried out in a similar way. For example, to create the squared value of the variable `age` in the NBA data set, we can put the name `agesq` in the Target Variable box, then in the type `age**2` in the Numeric Expression box, (the double asterisk translates to ‘raised to the power’).

Creating Dummy Variables

In some cases, we would like to create a dummy variable (also known as an indicator variable) to cover categories represented by a string variable (or a numeric variable representing categories) in our data set. For example, in our NBA data set we have a variable called ‘position’ which is a

string variable. For example, the entry ‘C’ stands for the position ‘Center’, ‘PF’ is ‘Power Forward’, and so on. We would like to create a dummy variable that equals 1 if a player is, say, a center, 0 otherwise. Similarly, we can create another dummy variable equal to 1 if a player is a power forward, 0 otherwise. We can do this for the other positions as well. To do this we start by clicking the the Transform option on the main menu, then choose Create Dummy Variables,



This will bring up a dialog box where we can select the variable that we want to convert to dummies. Click the variable position, then use the blue arrow key to move it into the Create Dummy Variables for: box. Under Main Effect Dummy Variables, the Create main-effect dummies box is already checked as the default, and that is what we want. Below this is a box for Root Names. Here we need to provide a root name (sometimes called a stub name) for the dummies that will be created. For example, we can put pos in this box. That means that the

dummies created will all have a name that begins with ‘pos’ and then will have an SPSS-generated remainder for the name. Lastly, in the box near the bottom left, titled Measurement Level Usage, we want to select Create dummies for all variables. Lastly, click OK,

Create Dummy Variables

Variables:

- player
- age
- salary
- team
- weight
- height
- games
- minspg
- nrebson

Dummy Variable Labels

☒ Use value labels

☐ Use values

Value Order

☒ Ascending

☐ Descending

Macros

☐ Omit first dummy category from macro definitions

Note: It is conventional to start macro names with !.

Measurement Level Usage

☐ Do not create dummies for scale variable values

☒ Create dummies for all variables

This dialog requires the Python Essentials

Create Dummy Variables for:

- position

Main Effect Dummy Variables

☒ Create main-effect dummies

Root Names (One Per Selected Variable):

pos

Macro Name:

Two-Way Interactions

☐ Create dummies for all two-way interactions

Root Name:

Macro name:

Three-Way Interactions

☐ Create dummies for all three-way interactions

Root Name:

Macro name:

OK Paste Reset Cancel Help

The output viewer shows the details of the dummy variables created,

Create dummy variables

Variable Creation

Label	
pos_1	position=C
pos_2	position=PF
pos_3	position=PG
pos_4	position=SF
pos_5	position=SG

We see that the variable pos_1 is for centers, pos_2 is for power forwards, etc. The Data View now shows five new columns, one for each position, with 1's indicating that the player plays the corresponding position,

pos_1	pos_2	pos_3	pos_4	pos_5
.00	1.00	.00	.00	.00
.00	.00	1.00	.00	.00
.00	.00	.00	1.00	.00
1.00	.00	.00	.00	.00
.00	.00	.00	.00	1.00
.00	1.00	.00	.00	.00
.00	.00	.00	.00	1.00
1.00	.00	.00	.00	.00
.00	.00	.00	1.00	.00
.00	.00	.00	1.00	.00
.00	.00	.00	.00	1.00

Note that switching to the Variable View tab we can see that the five dummies are defined under the Label column. We can also edit the Name of each variable to make it something more recognizable (e.g., changing pos_1 to center),

19	seasons	Numeric	2	0	
20	center	Numeric	8	2	position=C
21	power_F	Numeric	8	2	position=PF
22	point_G	Numeric	8	2	position=PG
23	pos_4	Numeric	8	2	position=SF
24	pos_5	Numeric	8	2	position=SG

This same process can be used to create dummy variables for an LSDV estimation (including year dummies) using panel data (see Chapter 7 in *Regression Basics*).

[**Note:** If the user wants to remove a variable from the data set, simply click the column containing the data to be removed, then either use a right-click, then choose clear, or choose Edit on the main menu and choose Clear.]

4. Descriptive Statistics and Correlations

Descriptive Statistics

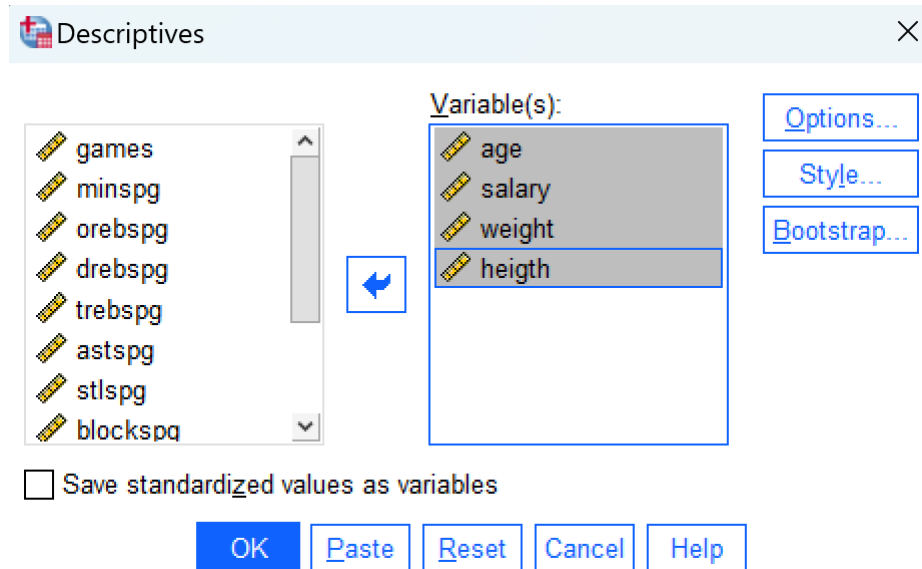
One of the first things that a researcher will compute once their data set has been read into SPSS are descriptive statistics (also called summary statistics). This will allow the researcher to get a feel for the values in the data set (e.g., how big are typical values, and how much they vary). It may also reveal unusual observations that will show up as minimum or maximum values. (In some cases, it may uncover errors in data. For example, if the minimum value for players' weight was -160, then this would indicate that there is at least one data point that was entered incorrectly into the data set since weight cannot be negative.)

To produce descriptive statistics, start on the Data View page, then on the main menu click on Analyze, then Descriptive Statistics, and then choose Descriptives...:

The screenshot shows the IBM SPSS Statistics Data Editor window with the file 'nba.sav [DataSet1]'. The 'Analyze' menu is open, and 'Descriptive Statistics' is selected. The 'Descriptives...' option is highlighted. The data table shows 21 NBA players and their statistics for games, minutes, and points.

Player	games	minspg	orebsp
1 Aaron Gordon	453	28.01250	1.562500
2 Aaron Holiday	182	18.40000	.200000
3 Aaron Nesmith	46	14.50000	.600000
4 Al Horford	881	32.17860	2.035710
5 Alec Burks	544	21.69230	.553846
6 Aleksej Pokusevski	45	24.20000	.700000
7 Alex Caruso	184	18.95000	.500000
8 Alex Len	531	17.48000	1.710000
9 Andre Iguodala	1192	31.65290	.894118
10 Andrew Wiggins	525	35.13750	1.250000
11 Anfernee Simons	154	15.03330	.266667
12 Anthony Edwards	72	32.10000	.800000
13 Anthony Gill	26	8.40000	.600000
14 Austin Rivers	588	24.24170	.316667
15 Avery Bradley	598	26.67860	.550000
16 Bam Adebayo	287	27.55000	2.075000
17 Ben McLemore	492	20.52220	.377778
18 Bismack Biyombo	702	20.03000	1.990000
19 Blake Griffin	668	33.00000	1.746150
20 Boban Marjanovic	263	9.65000	1.387500
21 Bogdan Bogdanovic	253	28.60000	.475000

This will bring up a dialog box where you can choose the variables for which descriptive statistics will be computed. Variable names can be highlighted by clicking them, and then clicking the blue arrow key will add them to the list in the Variable(s) box. Note that variables can be added one at a time, or by holding down the “Ctrl” key multiple variables can be highlighted and added together. Or one can click a variable on the list, and then hold down the “Shift” key and click on a variable further down the list and in doing so highlight all the variables between them. Then the blue arrow key can be used to add the block of variables to the list. Suppose we wish to compute descriptive statistics for the variables, age, salary, weight, and height. We would have the following,



Now the variables have been selected we hit the OK button and SPSS produces results that are displayed in the output window:

*Output1 [Document1] - IBM SPSS Statistics Viewer

File Edit View Data Transform Insert Format Analyze Graphs Utilities Extensions Window Help

Output

- Descriptives
 - Title
 - Notes
 - Descriptive Statistics

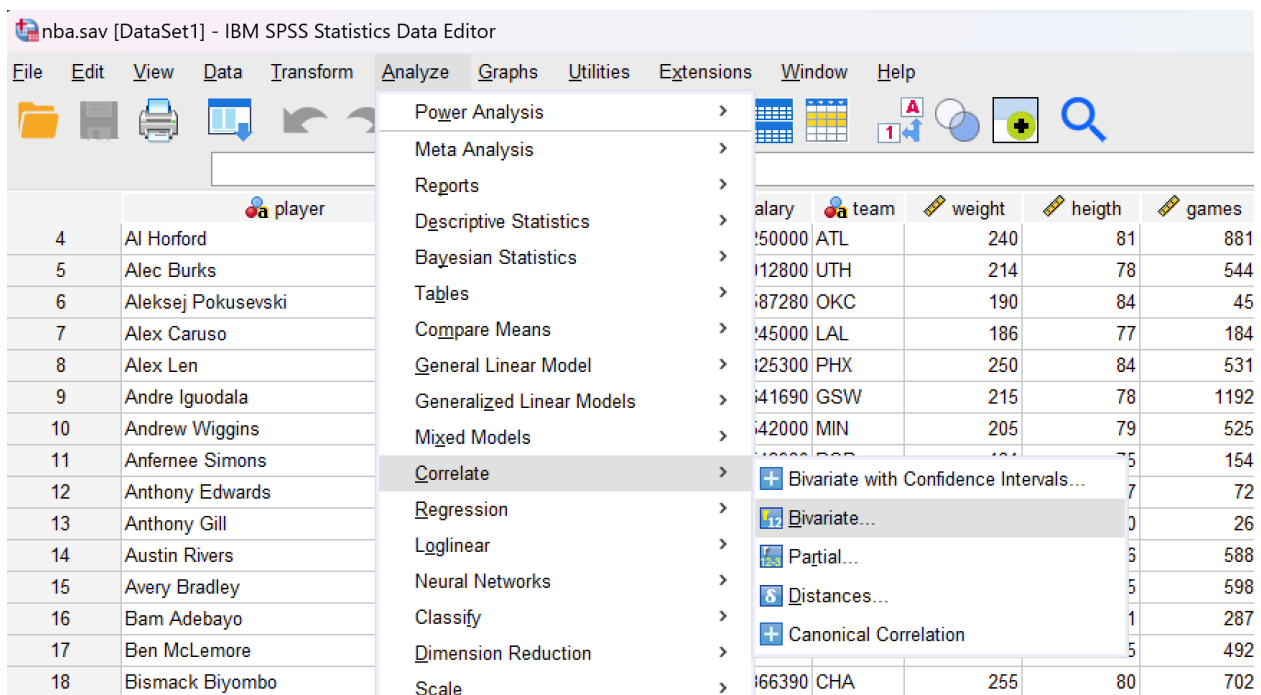
Descriptives

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
age	311	20	41	26.73	4.314
salary	311	.954267	44.066400	9.84867925	10.64244407
weight	311	164	290	214.51	23.766
heigh	311	72	87	77.98	2.988
Valid N (listwise)	311				

Correlations

In addition to descriptive statistics, a researcher may wish to study the correlation between various pairs of variables. This may be particularly useful when we suspect that high multicollinearity is a problem in our regression analysis, (see Chapter 8 in *Regression Basics*). Suppose we wish to compute the correlations between the variables age, weight, and height. This can be done in SPSS by clicking Analyze on the main menu, then click Correlate, then choose Bivariate... This will bring up a dialog box where you can choose which variables to include.



Choosing age, weight, and height by highlighting and using the blue arrow key, we next hit the OK button and we get results sent to the output page.

Note: In most cases the output shown on the output page can be highlighted with a click, then use a right click, select copy, and then the output can be pasted directly into an MS Word[®] document, an MS Excel[®] spreadsheet, or similar programs. Alternatively, while on the output window, you can go to File, choose Export..., and then follow the instructions for exporting the output from the viewer to a Word document or other formats, including Excel.

Correlations		age	weight	heigh
age	Pearson Correlation	1	.108	-.042
	Sig. (2-tailed)		.057	.462
	N	311	311	311
weight	Pearson Correlation	.108	1	.725**
	Sig. (2-tailed)	.057		<.001
	N	311	311	311
heigh	Pearson Correlation	-.042	.725**	1
	Sig. (2-tailed)	.462	<.001	
	N	311	311	311

** . Correlation is significant at the 0.01 level (2-tailed).

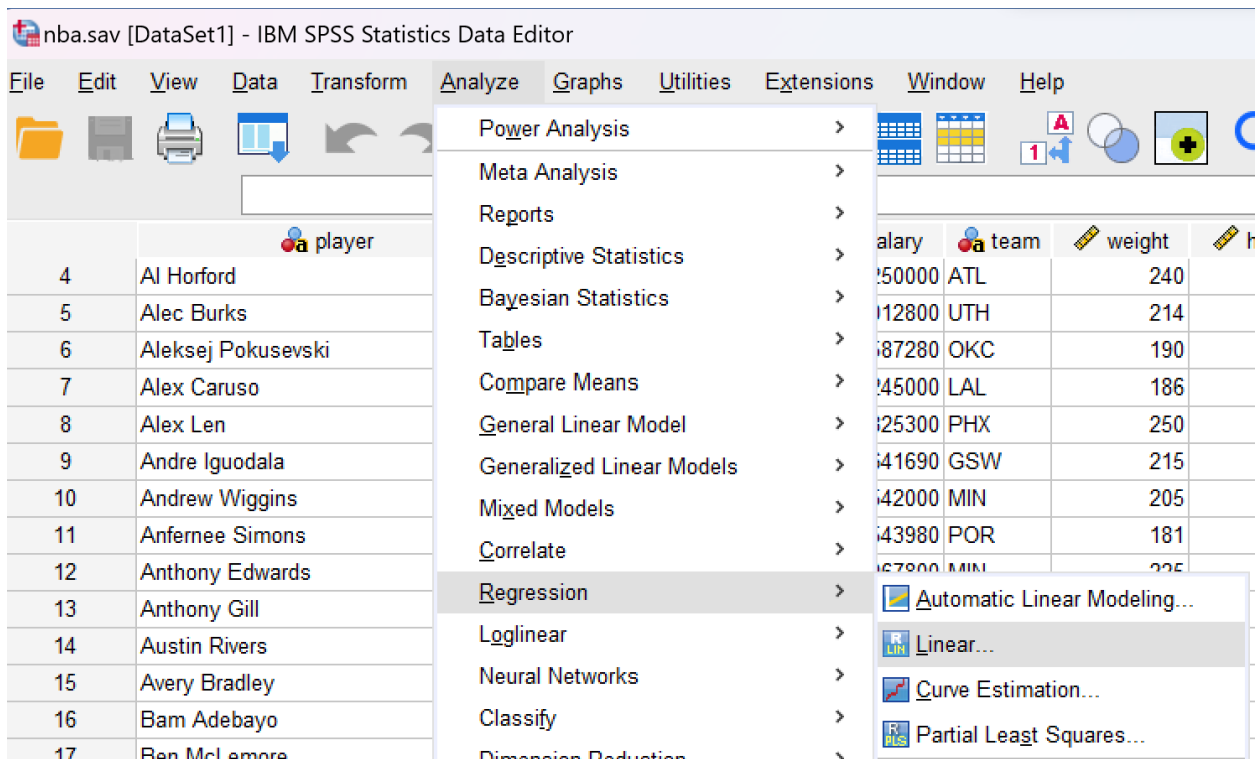
We see in the correlation matrix that there is a positive, significant correlation between weight and height (0.725), which one would expect.

5. Ordinary Least Squares (OLS) Regression

Estimating an OLS Regression and Saving the Predicted Values and Residuals

Suppose we wish to estimate an OLS regression using our NBA data set with salary as the dependent variable and seasons and points per game ('pointspg') as our independent variables.

We begin by choosing Analyze on the main menu, then Regression, then Linear...



This will bring up a dialog box where we can choose our Dependent variable by highlighting and using the blue arrow button. Next, we chose the Independent(s), seasons and pointspg, by highlighting and using the blue arrow to add them to the estimation.

Linear Regression

Dependent: salary

Block 1 of 1

Independent(s): seasons, pointspg

Method: Enter

Selection Variable: Rule...

Case Labels:

WLS Weight:


OK Paste Reset Cancel Help

Statistics... Plots... Save... Options... Style... Bootstrap...

position age team weight height games minspg orebspg drebspg trebspg astspg stlspg blockspg tovspg foulspg pointspg seasons

At this point, clicking the OK button would send OLS regression results to the output window.

Before doing so, however, we may want to save the predicted values for salary (the \hat{Y} 's) and the residuals from the regression (the e 's). This can be done by clicking on the “Save...” box on the right-hand side. This brings up a secondary dialog box where we can choose what to save. Click the “Unstandardized” predicted values and residuals.


Linear Regression: Save
×

Predicted Values

☒ Unstandardized
☐ Standardized
☐ Adjusted
☐ S.E. of mean predictions

Distances

☐ Mahalanobis
☐ Cook's
☐ Leverage values

Prediction Intervals

☐ Mean ☐ Individual
Confidence Interval: %

Residuals

☒ Unstandardized
☐ Standardized
☐ Studentized
☐ Deleted
☐ Studentized deleted

Influence Statistics

☐ DfBetas
☐ Standardized DfBetas
☐ DfFits
☐ Standardized DfFits
☐ Covariance ratios

Coefficient statistics

☐ Create coefficient statistics
☒ Create a new dataset
Dataset name:
☐ Write a new data file

Export model information to XML file

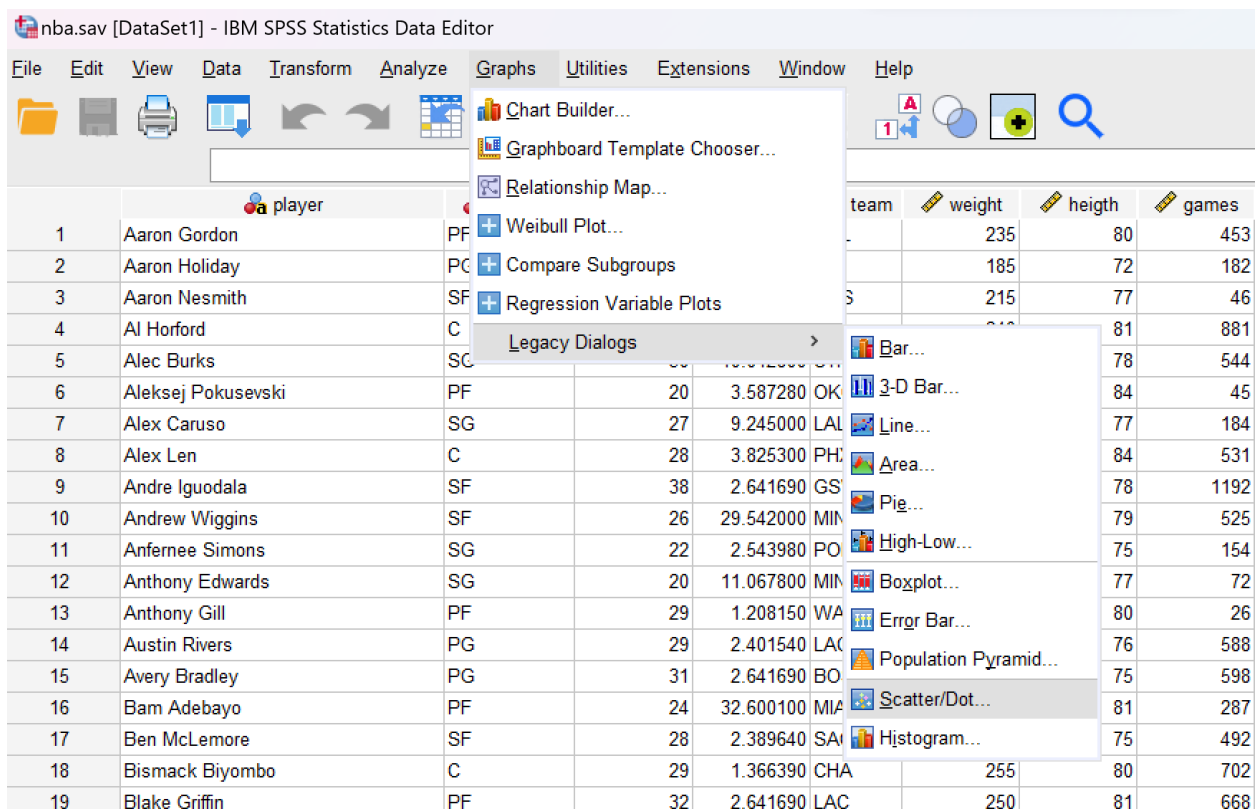
☒ Include the covariance matrix

Hitting the Continue button will return us to the Linear Regression dialog box. We can now click OK, and the regression results will be sent to the output window, and the predicted values for salary and the residuals will be added to our data set on the Data View page with variable names PRE_1 and RES_1, respectively.

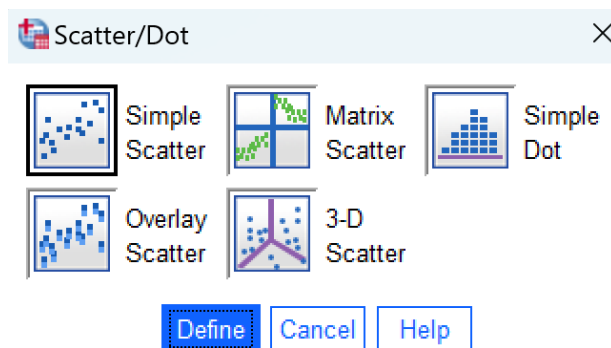
6. Creating a Scatterplot with Reference Line

It may be useful to create a scatterplot for a pair of variables and include a reference line in the graph. This is a simple way of visually understanding the relationship between two variables, and it may also be helpful for visually checking for heteroskedasticity (see Chapter 8 in *Regression Basics*).

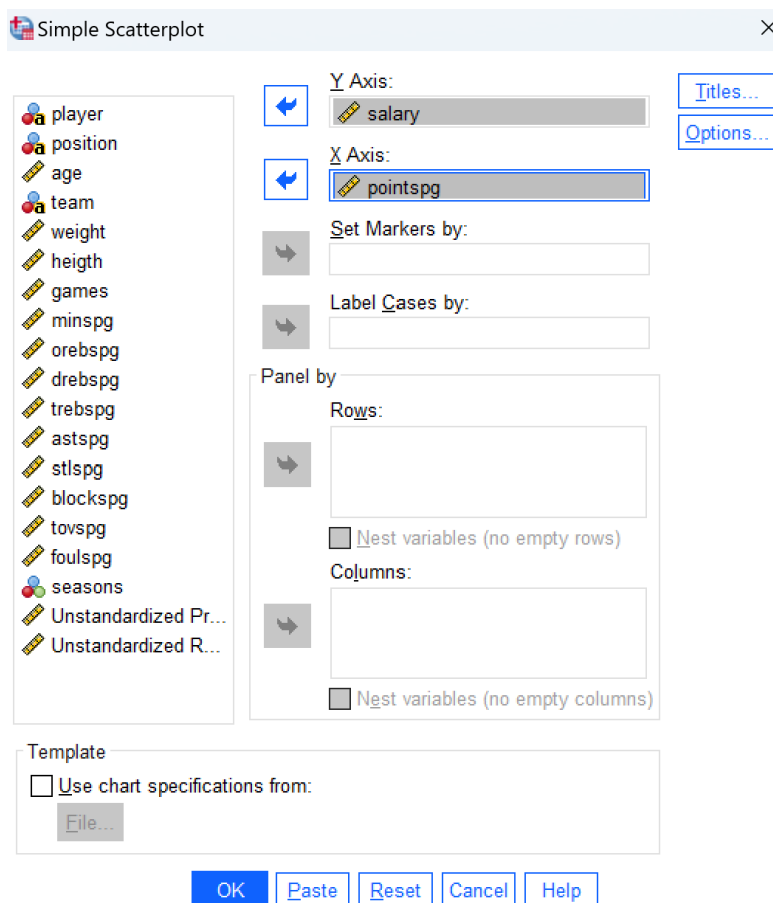
Creating scatter plots in SPSS can be done in several ways. One is to work with the “Legacy Dialogs”. This is done by clicking on Graphs on the main menu, then choose Legacy Dialogs, and then Scatter/Dot...



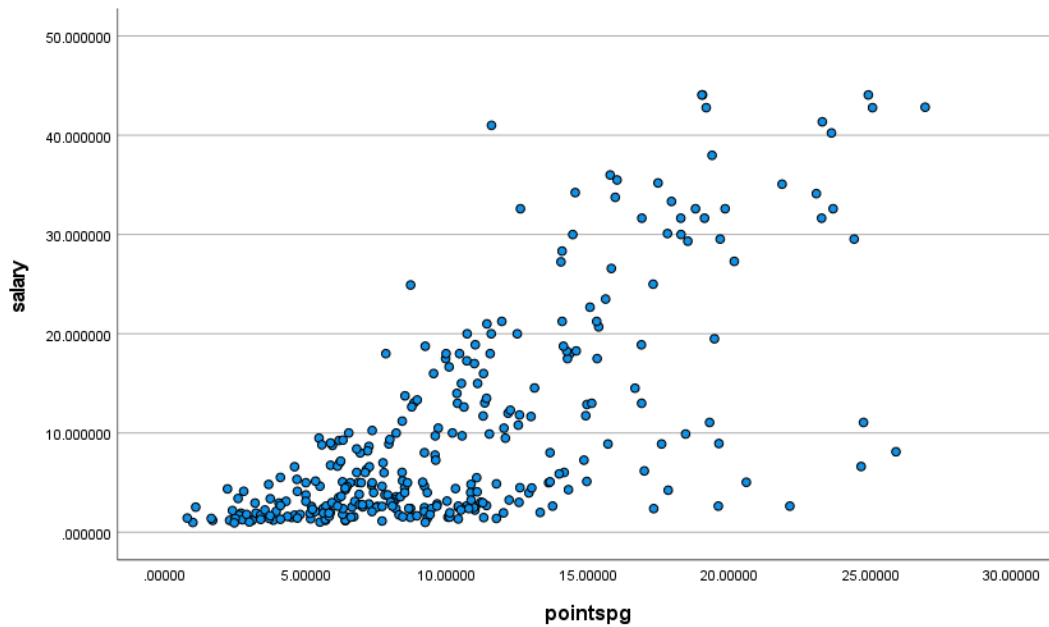
This will bring up a smaller dialog box with options,



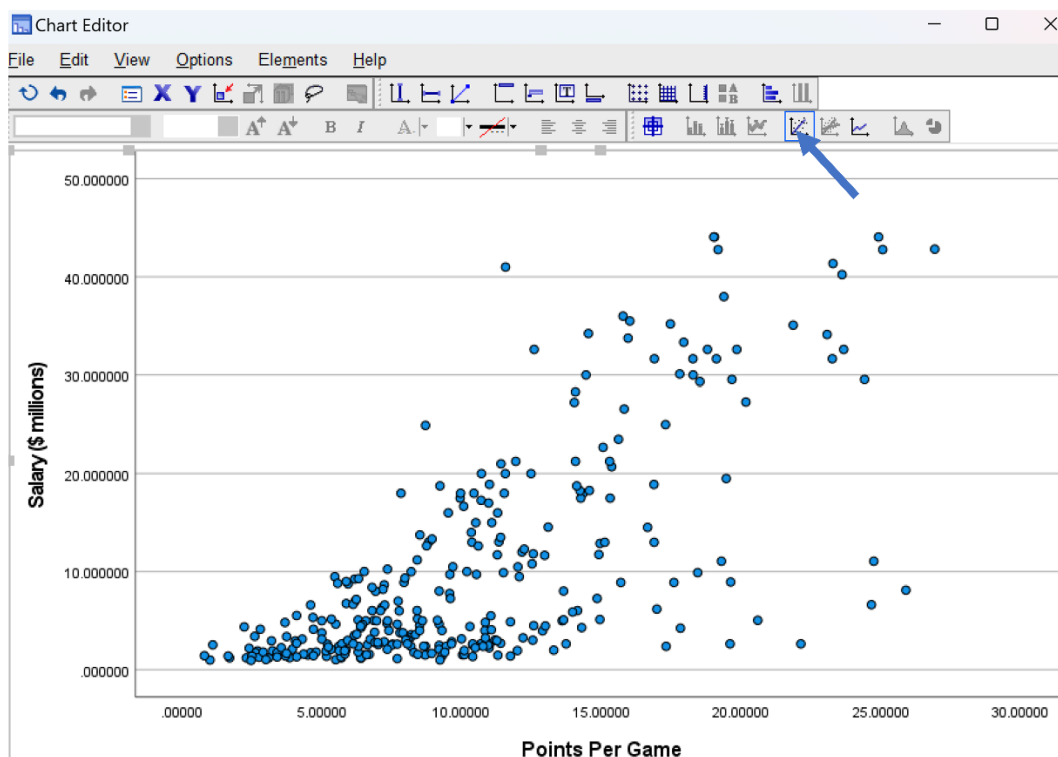
Choose Simple Scatter, then click Define. This will bring up a dialog box where you can choose which variables to plot on the Y and X axis. Suppose we wish to plot salary on the Y axis and points per game (pointspg) on the X axis. Using the blue arrow keys, we can select these two variables, then choose OK,



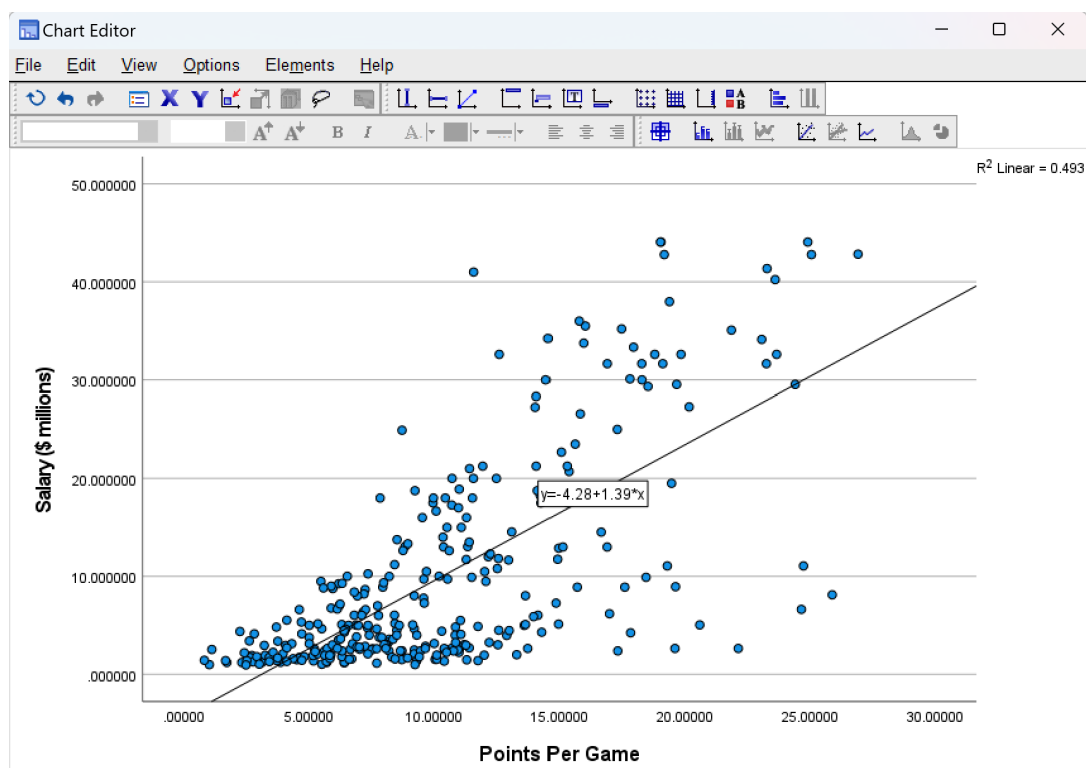
The scatter plot will be sent to the output page,



Customizing the scatterplot can be done by double clicking the graph in the output viewer. This brings up the Chart Editor which shows the current graph. By clicking on various elements of the graph, you can customize things. For example, we could change pointspg to Points Per Game by clicking this axis label and replacing the current label. We can do the same for salary, making it Salary (\$ millions). In addition, we can add a best fit line (i.e., OLS regression line) to the chart by selecting the icon on the ribbon showing a line fitted to a scatterplot (see the blue arrow in the graph below),



The new graph with contain the best fit line, including the regression equation and the R^2 ,



Another useful scatterplot is one where the residuals from a regression estimation are plotted against the predicted (also known as fitted) values. As discussed in Chapter 8 of *Regression Basics*, this is a common way to visually check for heteroskedasticity. Since we saved the predicted and residual values from our previous regression, we can proceed as before:

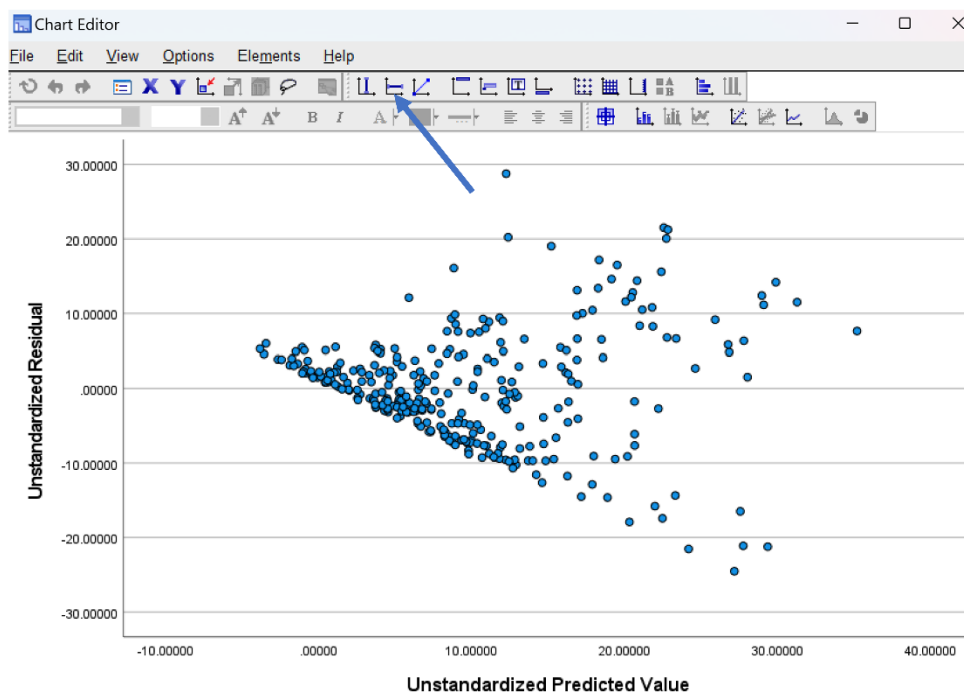
Click on Graphs, Legacy Dialogs, Scatter/Dot...

Select Simple Scatter, then click Define

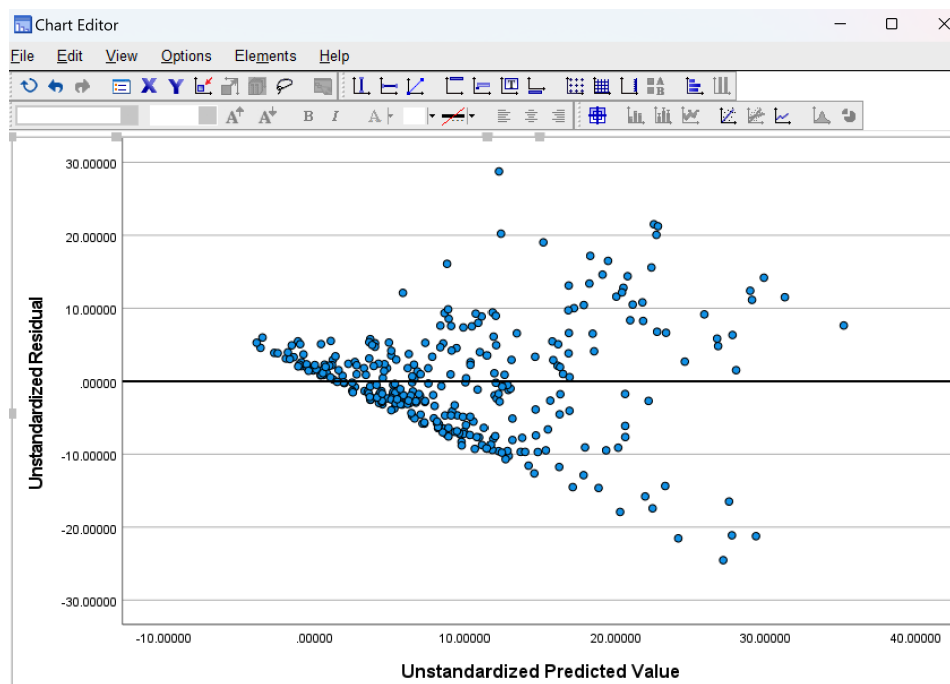
Select the variable RES_1 for the Y axis, and PRE_1 for the X axis.

Select OK

This will send the scatterplot to the output window. Clicking the graph twice will launch the Chart Editor. To add a horizontal line at zero, we can click the icon showing a horizontal reference line, (see the blue arrow),



A reference line is now placed at zero on the vertical axis,



As is evident from the graph, we appear to have heteroskedasticity as the spread of the dots gets wider as the predicted value increases.

An alternative to the Legacy Dialogs, graphs can be built by choosing Graphs, then Chart Builder.... This will launch a graph building dialog box where chart types and variables can be dragged into place, and then click OK. This will send the graph to the output window. Double clicking the graph will once again launch the Chart editor.

Once a graph is completed, it can be right-clicked and copied from the output window into a Word document or elsewhere (or using the File, Export... command will allow the user to save the graph in the desired format and location).

7. Breusch-Pagan Test for Heteroskedasticity and OLS with Robust Standard Errors

As discussed in Chapter 8 of *Regression Basics*, the presence of heteroskedasticity creates problems as the regular OLS coefficient standard errors used for inference tests are not correct. We saw in our scatterplot that there seems to be evidence of heteroskedasticity in the NBA salary regression. To be more careful about our conclusion, we can conduct the Breusch-Pagan test for heteroskedasticity, and if present then we can use robust standard errors to fix the problem for the standard errors and allow for legitimate inference tests. As a benchmark, below is the simple OLS regression results for our NBA data set with salary as the dependent variable, and seasons and points per game (pointspg) as our independent variables,

Coefficients^a

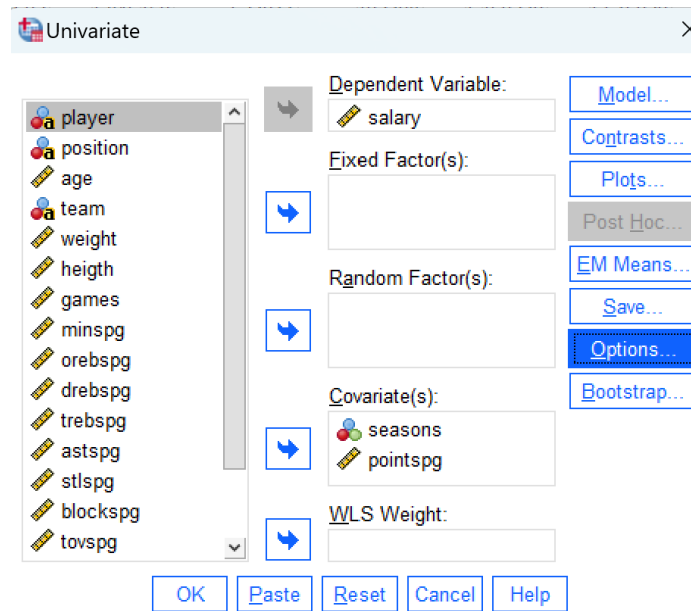
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-5.188	.963		-5.389	<.001
	seasons	.300	.103	.125	2.925	.004
	pointspg	1.299	.084	.658	15.392	<.001

a. Dependent Variable: salary

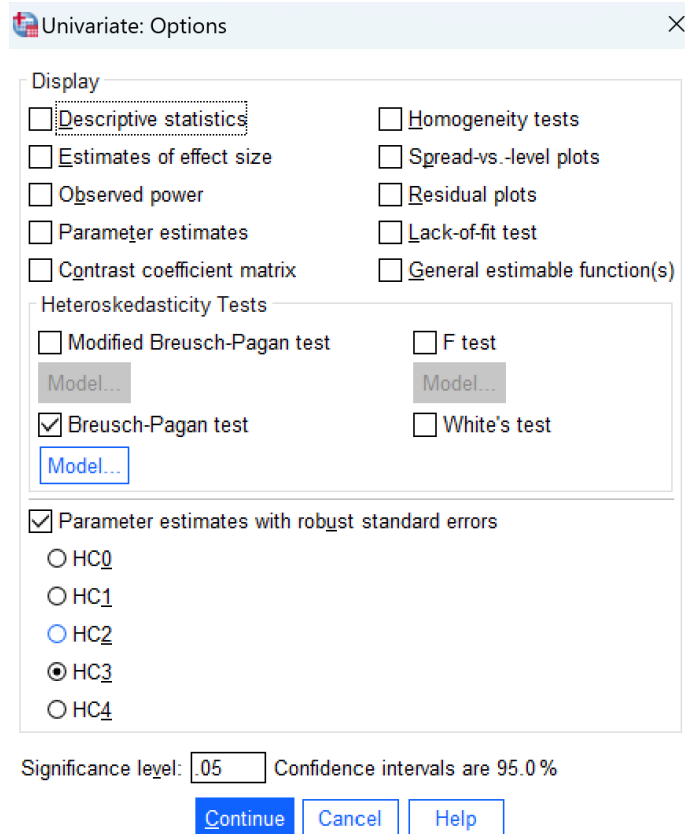
To have SPSS produce the Breusch-Pagan test and then estimate the regression with robust standard errors, we choose Analyze from the main menu, then choose, General Linear Model, and then Univariate...,

The screenshot shows the IBM SPSS Statistics Data Editor interface. The main window displays a dataset named 'nba.sav [DataSet1]' with variables 'games', 'minspg', 'astspg', 'stlspg', 'blockspg', and 'tovspg'. The 'Analyze' menu is open, showing the path: Analyze > General Linear Model > Univariate... The 'Univariate...' option is highlighted. The 'Data Editor' window shows the first 19 rows of data, with the first row having values: 1, 30, 453, 28.01250, 2.537500, .725000, .650000, 1.512500.

This will bring up a dialog box that allows the user to choose the dependent variable salary. The independent variables are placed in the Covariate(s) box.



Choosing the Options.... button brings up a sub-dialog box where the user can choose the Breusch-Pagan test and Parameter estimates with robust standard errors,



The image shows the 'Univariate: Options' dialog box in SPSS. The 'Display' section has several checkboxes: 'Descriptive statistics' (checked), 'Estimates of effect size', 'Observed power', 'Parameter estimates', 'Contrast coefficient matrix', 'Homogeneity tests', 'Spread-vs.-level plots', 'Residual plots', 'Lack-of-fit test', and 'General estimable function(s)'. The 'Heteroskedasticity Tests' section has 'Modified Breusch-Pagan test' (unchecked), 'Breusch-Pagan test' (checked), 'F test' (unchecked), and 'White's test' (unchecked). Each test has a 'Model...' button. The 'Parameter estimates with robust standard errors' section is checked, with radio buttons for HC0, HC1, HC2, HC3 (selected), and HC4. At the bottom, the 'Significance level' is .05 and 'Confidence intervals are 95.0 %'. There are 'Continue', 'Cancel', and 'Help' buttons.

Clicking Continue, then OK sends the Breusch-Pagan test results and the OLS regression with robust standard errors to the output windows,

Breusch-Pagan Test for Heteroskedasticity ^{a,b,c}		
Chi-Square	df	Sig.
145.284	1	.000

a. Dependent variable: salary

b. Tests the null hypothesis that the variance of the errors does not depend on the values of the independent variables.

c. Predicted values from design: Intercept + seasons + pointspg

The Chi-Square statistic (145.284) and the associated Sig. value (0.000) strongly rejects the null hypothesis of homoskedasticity. The regression output with robust standard errors is,

Parameter Estimates with Robust Standard Errors

Dependent Variable: salary

Parameter	B	Robust Std.	t	Sig.	95% Confidence Interval	
		Error ^a			Lower Bound	Upper Bound
Intercept	-5.188	.915	-5.672	<.001	-6.987	-3.388
seasons	.300	.120	2.504	.013	.064	.535
pointspg	1.299	.122	10.678	<.001	1.059	1.538

a. HC3 method

The new output has the same parameter estimates as the benchmark regression above (which is expected), but the standard errors have now been corrected (using the default HC3 method) allowing for legitimate inference tests for the estimated parameters.

8. Testing for Multicollinearity: Variance Inflation Factor (VIF)

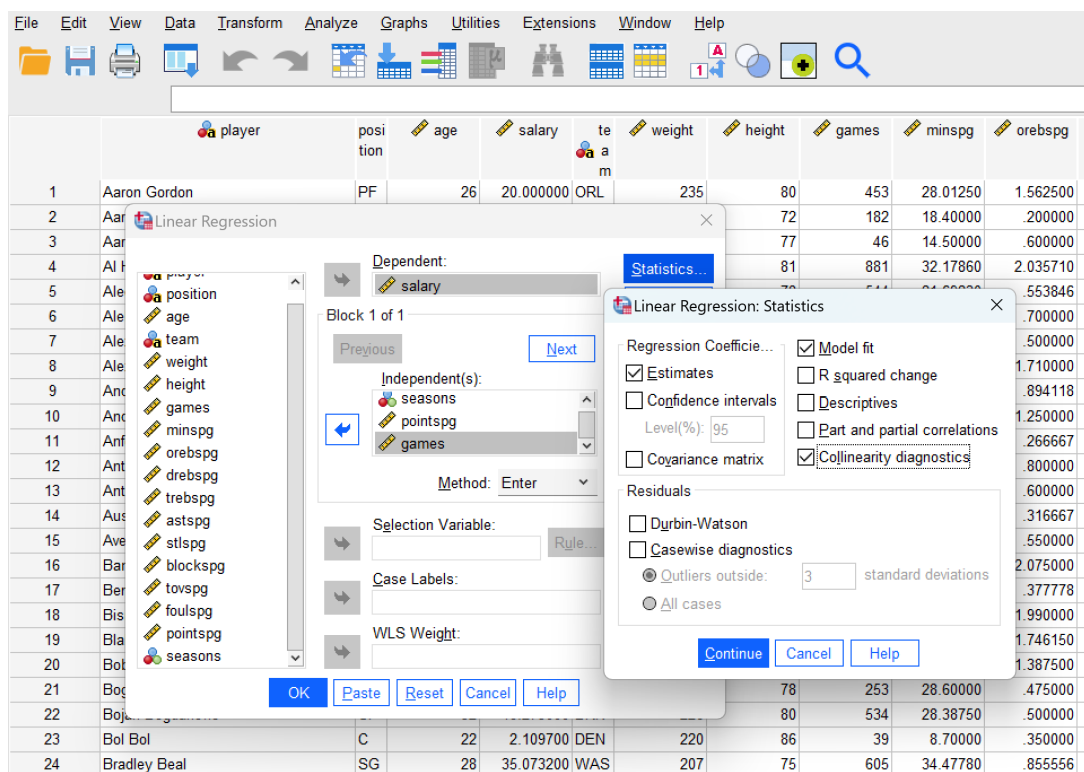
As discussed in Chapter 8 of *Regression Basics*, high multicollinearity can be a nuisance in multiple regression models. The variance inflation factor (VIF) is a common measure used to test for high multicollinearity. To illustrate how to produce VIF estimates with SPSS, we will add an additional independent variable to our NBA salary model, namely ‘games’, which is the career number of games a player has played in the NBA. This variable would likely be highly correlated with ‘seasons’, which is already in our regression model. Running a simple OLS regression we find,

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-3.818	1.054		-3.621	<.001
	seasons	-.724	.356	-.302	-2.034	.043
	pointspg	1.176	.093	.596	12.688	<.001
	games	.018	.006	.468	2.999	.003

a. Dependent Variable: salary

Notice that, while all three estimated coefficients have a p-value of less than 0.05, suggesting that they are statistically significant, the coefficient to seasons is unexpectedly negative. [Note that this regression also suffers from heteroskedasticity, but we will ignore this issue for now.] An unexpected sign is one of the indicators that we may have high multicollinearity present in the model. We can re-estimate the model, but this time click on the ‘Statistics...’ box, and then check the ‘Collinearity diagnostics’ box as seen in the picture below,



Hitting ‘Continue’, then ‘OK’ to run the regression we get the same regression results, but now a VIF column has been added to the output (as well as a Collinearity Tolerance column, equal to $1/\text{VIF}$),

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Collinearity Statistics	
		B	Std. Error	Beta			Tolerance	VIF
1	(Constant)	-3.818	1.054		-3.621	<.001		
	seasons	-.724	.356	-.302	-2.034	.043	.071	14.123
	pointspg	1.176	.093	.596	12.688	<.001	.708	1.413
	games	.018	.006	.468	2.999	.003	.064	15.575

a. Dependent Variable: salary

We see that both seasons and games have VIF values greater than our benchmark value of 10, indicating high multicollinearity.

9. Testing for Autocorrelation and Prais-Winsten Estimation

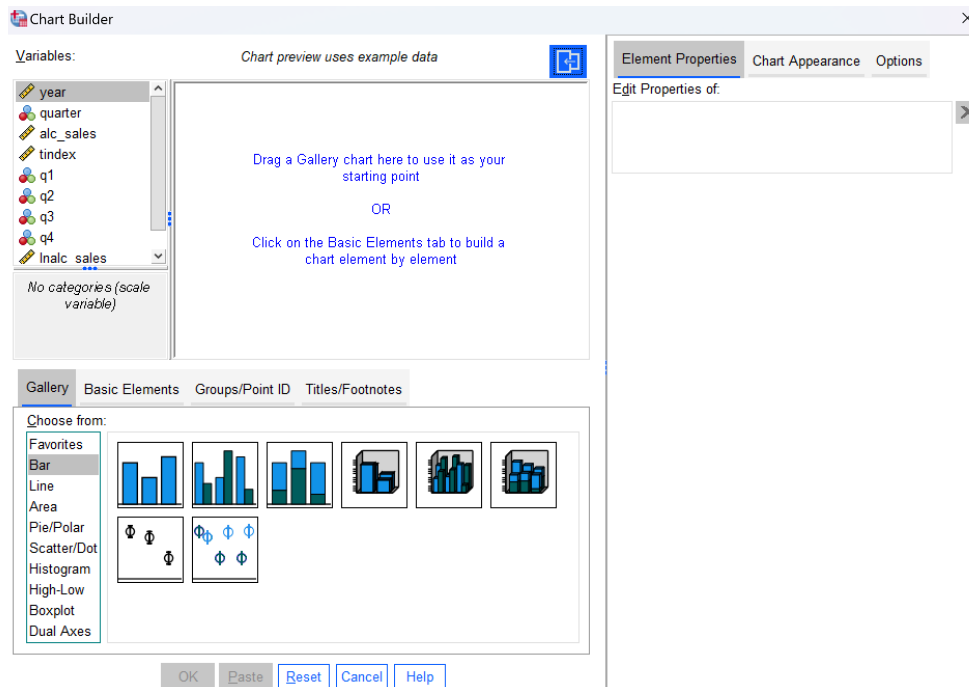
Autocorrelation, (also known as serial correlation) in the error structure is a common problem with time series data. When present the variance of the residuals (the e_t 's) tends to underestimate the variance of the true population's errors (the u_t 's). Since the standard errors of the parameter estimates are based on the variance of the residuals, this means the t statistics and their associated p-values are invalid. To illustrate how to test and correct for autocorrelation we will work with our alcohol beverages sales example using the “alc_sales.csv” data set (available on the companion website for *Regression Basics*). Our sample regression model was, (see Chapter 7),

$$\ln alc_sales_t = a + b_1(tindex) + b_2q2_t + b_3q3_t + b_4q4_t + e_t$$

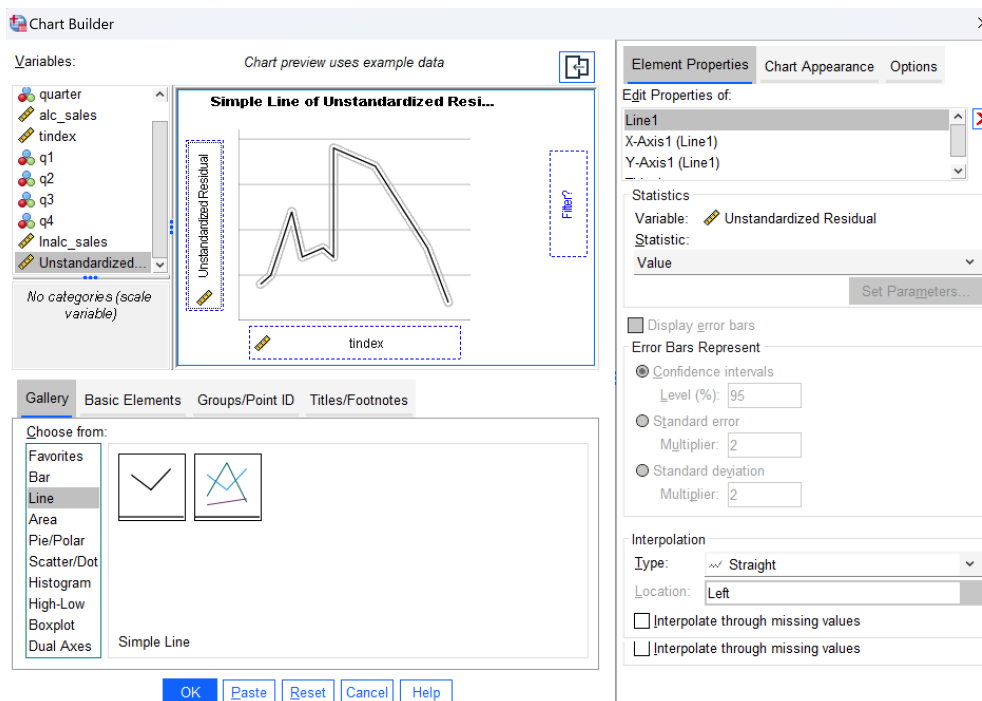
Where $\ln alc_sales_t$ is the natural log of alcoholic beverages sales in the U.S., $tindex$ is a time index, and $q2_t$, $q3_t$, and $q4_t$, were quarter dummies to pick up seasonal effects.

Graphical Detection of Autocorrelation

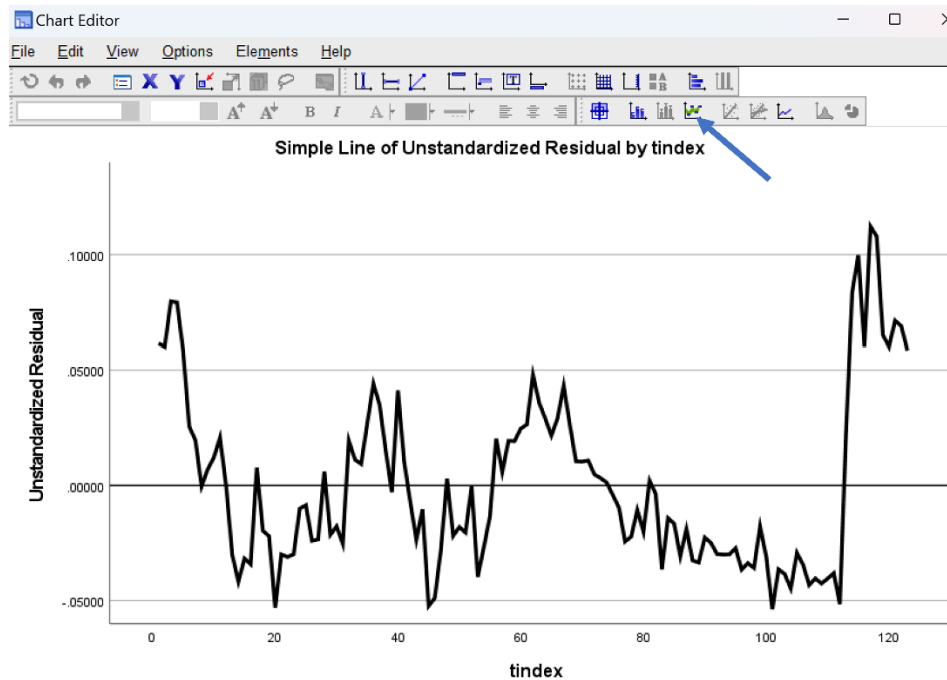
As discussed in Chapter 8, one method of detecting autocorrelation is to plot the errors from an OLS regression and check for a pattern. This can be done by estimating the above regression, saving the residuals, and then creating a connected line graph of the residuals over the time periods. We will do this with the Chart Builder this time (rather than the Legacy Dialogs) under the Graph menu. Select Graph on the main menu, then Chart Builder...



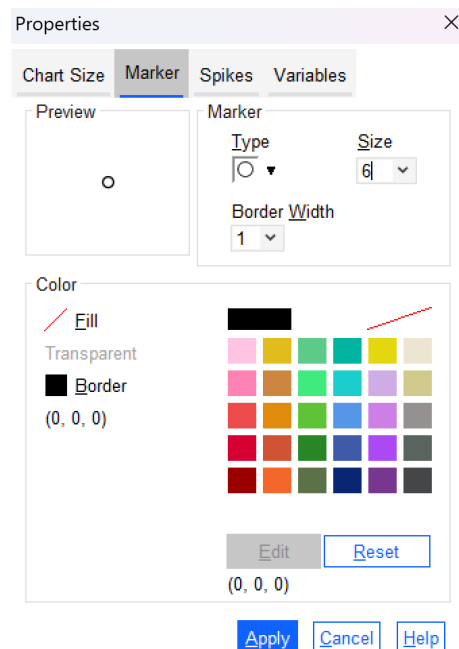
This brings up a dialog box, where we can choose Line from the Gallery box, and then drag the Simple version to the preview area and a preview of the graph is displayed. Next, we can drag tindex from the Variables box to the X-Axis, and the Unstandardized Residuals (RES_1) to the Y-Axis,



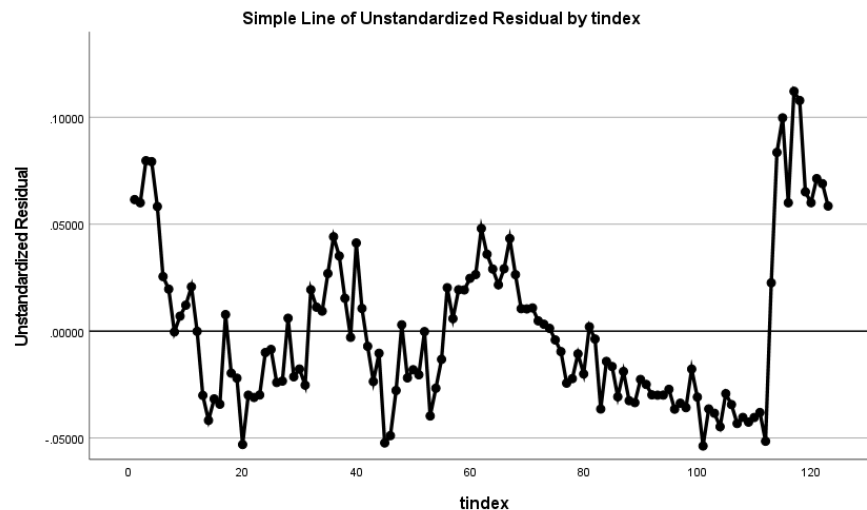
Clicking OK will send the graph to the output window. As before, double clicking the graph in the output window will start the Chart Editor where we can add a reference line at zero.



Clicking the icon with a chart with markers (see the blue arrow above), will allow you to add markers for the observations whose appearance are customizable.

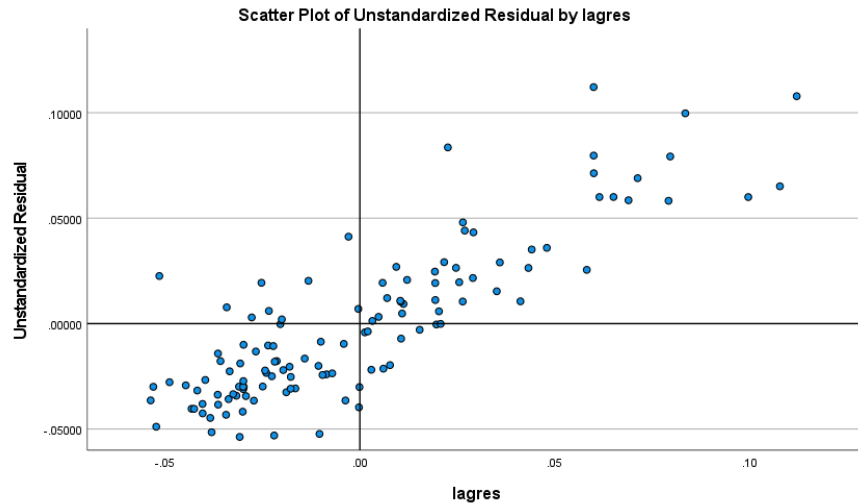


For example, choosing a size of 6, and black fill, then clicking Apply yields the following graph,



Viewing the graph, (which is the same as that shown in Figure 8.6a in *Regression Basics*) we see evidence of positive autocorrelation, with long strings of positive errors, then long strings of negative errors. In fact, from periods 82 to 112 we have thirty-one negative errors in a row – this is clearly *not* a random pattern.

An alternative graph was one where we had the current value of the residuals on the Y-Axis, and the lagged value of the residuals on the X-Axis, (see Figure 8.6b in *Regression Basics*). This can be done by first selecting Transform on the main menu, then selecting Compute... In the Target Variable box, we can add the name lagres (for lagged residual). In the Numeric Expression box we can use the command, lag(RES_1). Selecting OK will create a new column of data headed lagres which will contain the lagged values of the RES_1. From here we can create a simple scatter plot with RES_1 on the Y-Axis and lagres on the X-Axis and add reference lines at zero for both axes. The resulting graph appears below,

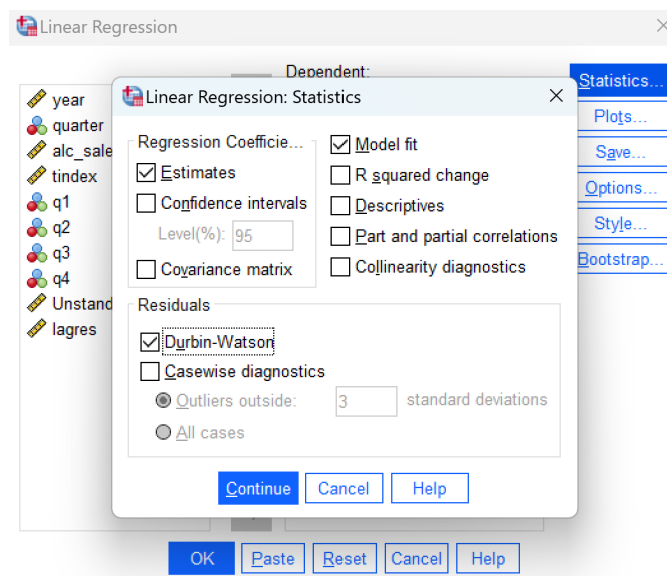


We see that the general pattern of upward sloping dots suggests positive autocorrelation.

Durbin-Watson Test of Autocorrelation

As described in Chapter 8, the Durbin-Watson statistic is often used to detect autocorrelation.

SPSS can produce this statistic when estimating a simple linear regression model. Start by going to Analyze, Regression, Linear..., enter the `ln_alcsales` for the Dependent variable, and `tindex`, `q2`, `q3`, and `q4` as the Independents. Then choose Statistics..., select Durbin-Watson under Residuals



Click Continue, then OK. The regression results will be sent to the output window and will include the Durbin-Watson statistic as part of the “Model Summary” box,

Model Summary ^b					
Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.995 ^a	.990	.989	.0380625008	.280

a. Predictors: (Constant), q4, tindex, q2, q3

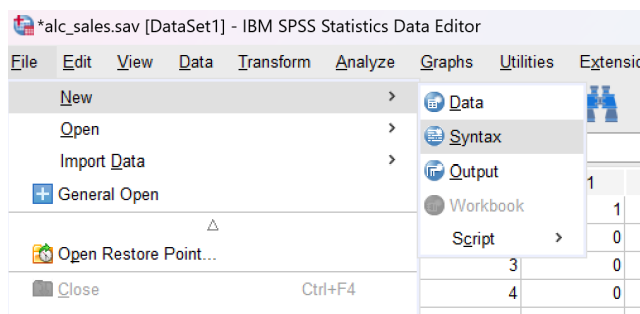
b. Dependent Variable: lncalc_sales

The value of 0.28 for the Durbin-Watson statistic, when compared to the appropriate Durbin-Watson table (see Chapter 8 in *Regression Basics*), indicates evidence of positive correlation.

Prais-Winsten Estimation

One possible solution to the problem of autocorrelation is to use the Prais-Winsten estimator.

This estimator uses an iterative approach to estimate the autocorrelation coefficient ($\hat{\rho}$) and then using this estimate transforms the model, hopefully purging it of autocorrelation, and then using OLS on the transformed model. This procedure can be carried out in SPSS by creating a syntax file with the appropriate commands, and then running that file. We begin by going to File on the main menus, then choosing New, and then Syntax,



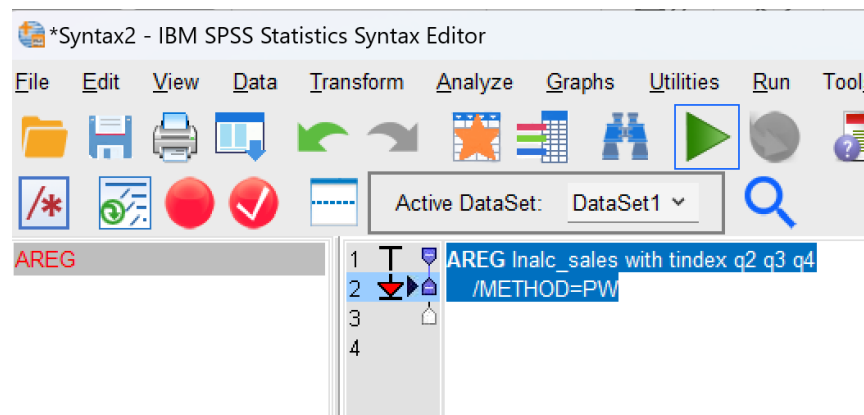
This will open a syntax window where we can type in the commands for estimating an “AREG” or AR(1) model where the current period’s error term is assumed to be a linear function of the previous period’s error term: $u_t = \rho u_{t-1} + \varepsilon_t$, with: $-1 < \rho < 1$

The following commands should be entered into the syntax window:

```
AREG lnc_sales with tindex q2 q3 q4  
/METHOD=PW
```

The first line calls the AREG estimator for the dependent variable lnc_sales, includes the word ‘with’, and then lists the independent variables tindex, q2, q3 and q4. The second line chooses the Prais-Winsten (PW) method for estimating the model.

Next, we highlight all the code, then click the green arrow button to run the code,



The results are then sent to the output window,

Iteration History

	Rho (AR1)		Durbin-Watson	Mean Squared Errors
	Value	Std. Error		
0	.839	.050	1.981	.000
1	.845	.049	1.996	.000
2 ^a	.845	.049	1.997	.000

The Prais-Winsten estimation method is used.

a. The estimation terminated at this iteration, because all the parameter estimates changed by less than .001.

Model Fit Summary

R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
.992	.984	.983	.020	1.997

ANOVA

	Sum of Squares	df	Mean Square
Regression	2.738	4	.685
Residual	.045	117	.000

Regression Coefficients

	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
tindex	.010	.000	.415	35.153	.000
q2	.118	.003	.513	35.476	.000
q3	.138	.004	.599	36.004	.000
q4	.246	.003	1.057	73.496	.000
(Constant)	8.377	.021		402.005	.000

The Prais-Winsten estimation method is used.

The results above (which are an edited version of the complete output) show that it took two iterations to settle down on an estimate of $\hat{\rho} = 0.845$, which was used for transforming the model

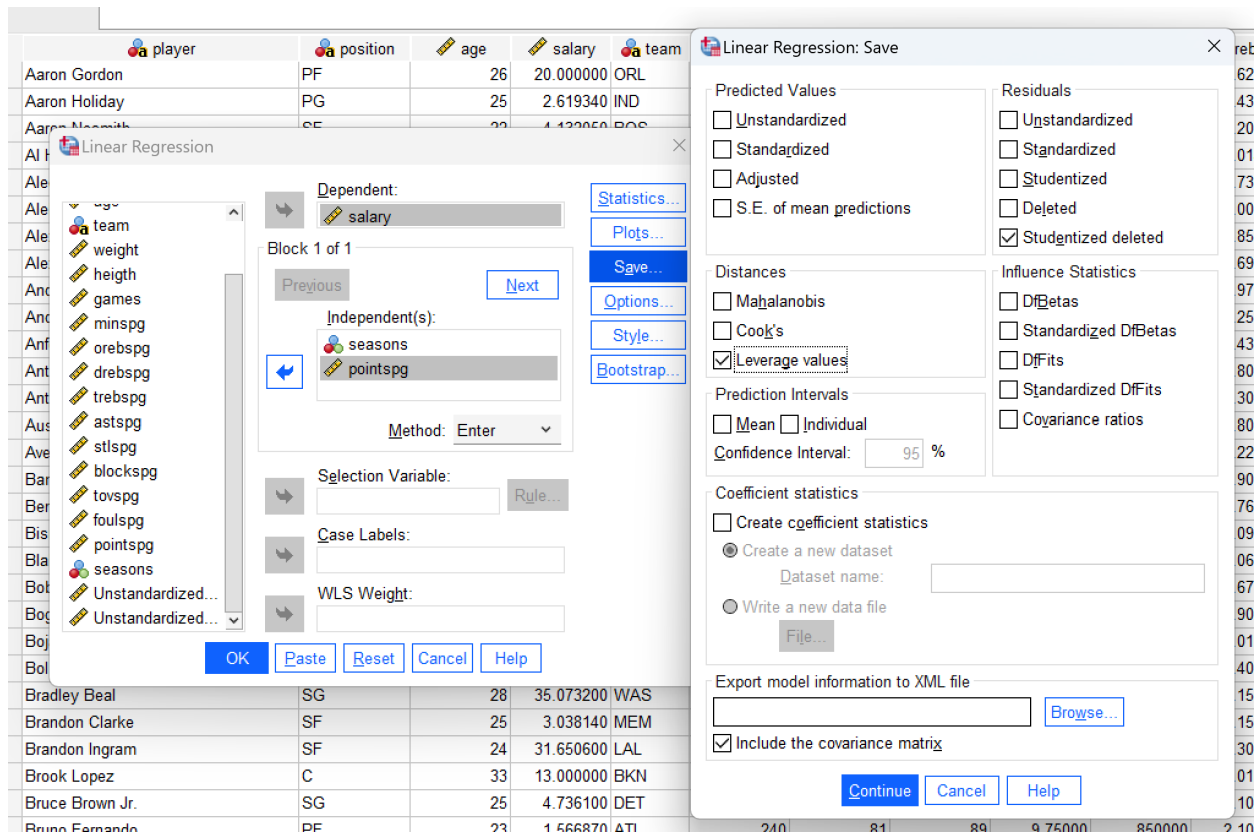
and estimating the final results (which match those presented in Chapter 8 or *Regression Basics*). Note that the syntax file can be saved for later use if desired.

10. Studentized Residuals and Leverage

In Chapter 8 of *Regression Basics* there is a discussion on influential observations. These are observations that contain the ingredients for having a strong impact on the coefficients of a regression estimation. The two main ingredients are being a strong outlier and having strong leverage. The measure called “studentized residuals” was used to detect strong outliers. This measure essentially looks for observations that have unusually large residuals (e_i 's). A studentized residual greater than 3 in absolute terms generally indicates a strong outlier.

Regarding strong leverage, as noted in Chapter 8, this essentially means that an observation's value for one or more of the independent (X_i) variables is unusually large or small, compared to the other observations. In cases where the leverage value for an observation is more than 3 times the mean value for leverage for all the observations, these may indicate strong leverage.

Both studentized residuals and leverage measures may be saved when performing an OLS regression in SPSS. For example, using the NBA data, we can use the main menu to choose Analyze, Regression, Linear..., then choose salary as the dependent variable, and seasons and points per game (pointspg) as the independent variables. Before running the regression, choose Save..., and select Studentized deleted, and Leverage values,

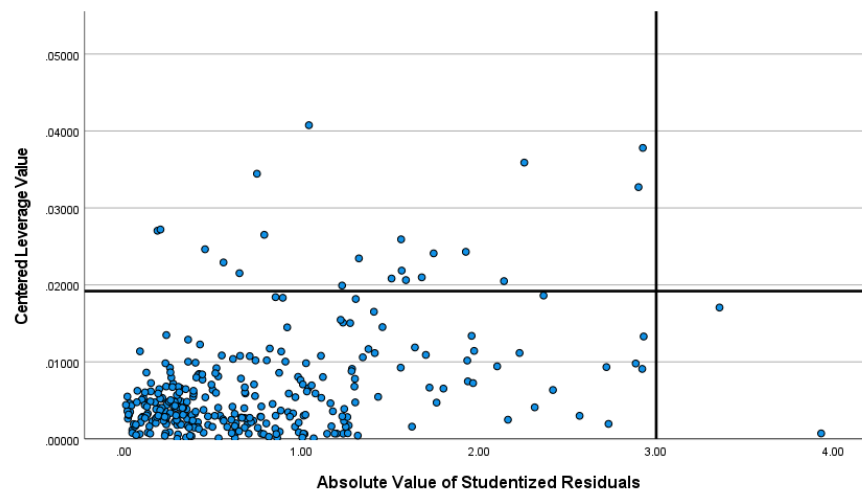


Next, click Continue, then OK to run the regression. This will send the regression results to the output window, and two new columns will be created in the data view window. One will be called SDR_1, which are the studentized residuals. The other will be called LEV_1, which are the leverage values. [Note that SPSS computes ‘centered’ leverage measures. These will differ from other programs, such as Stata and R, that report uncentered leverage measures.¹]

Using our ‘rule of thumb’ values for studentized residuals and leverage values, we can look for observations where their absolute value of SDR_1 is greater than or equal to 3, and their LEV_1 value is greater than 3 times the mean value for LEV_1. To compute the absolute value of

¹ It can be shown that the mean value for leverage is equal to $(k + 1)/n$, where k is the number of independent variables (X_i 's) in the model and n is the sample size. SPSS apparently centers the leverage members by removing the $1/n$ part of this equation, thus the mean value for SPSS centered leverage measures is simply k/n .

SDR_1, we can select Transform on the main menu, then choose Compute Variable. Next we provide a Target Variable name, such as abs_sdr (short for absolute value of studentized residual), and in the Numeric Expression box we can type, $\text{abs}(\text{SDR}_1)$, which will compute the absolute value of SDR_1. Hitting OK produces a new column of data headed abs_sdr with the absolute value of the studentized residuals. As for the leverage values, the mean is about 0.0064, so 3 times this mean is 0.0192. Lastly, we can create a scatterplot that has the absolute value of the studentized residuals on the horizontal axis, and the leverage values on the vertical axis. We can also place a reference line at 3 on the horizontal axis and at 0.0192 on the vertical axis and look to see if any observations exceed both these values. The resulting scatterplot is,



None of the observations in our NBA data set reach both thresholds, (though several come close).