

Occam's Pocket Knife

- [Home](#)
- [About Occam's Pocket Knife](#)
-

[Home](#) > [Uncategorized](#) > Translating SPSS to R: Simple ANOVA

Translating SPSS to R: Simple ANOVA

2012.10.19 [sunbyrne](#) [Leave a comment](#) [Go to comments](#)

This is where things start to get more interesting. The ANOVA is one of the older tools in the experimental psychology toolbox, and as a result there is quite a lot of depth here. I'm going to cover ANOVA in multiple pieces. Today's entry is the simplest case: between-subjects one-way ANOVA. I will discuss the basic procedures, unequal variance adjustment, contrasts, and posthocs.

Later entries will cover balanced factorial between-subjects ANOVA, unbalanced factorial between-subjects ANOVA, simple repeated-measures ANOVA, factorial repeated-measures ANOVA, and ultimately factorial mixed (that is, between- and within-subjects) ANOVA. One step at a time, though.

The simplest complication from a standard two-groups between-subjects *t*-test is a oneway ANOVA with three groups, so this is where I take my class. The example data set ([synchro.xls](#)) has two treatment conditions ("fast" and "slow") and a control condition; the dependent variable is the number of words recalled in some kind of memory test (the details aren't that important).

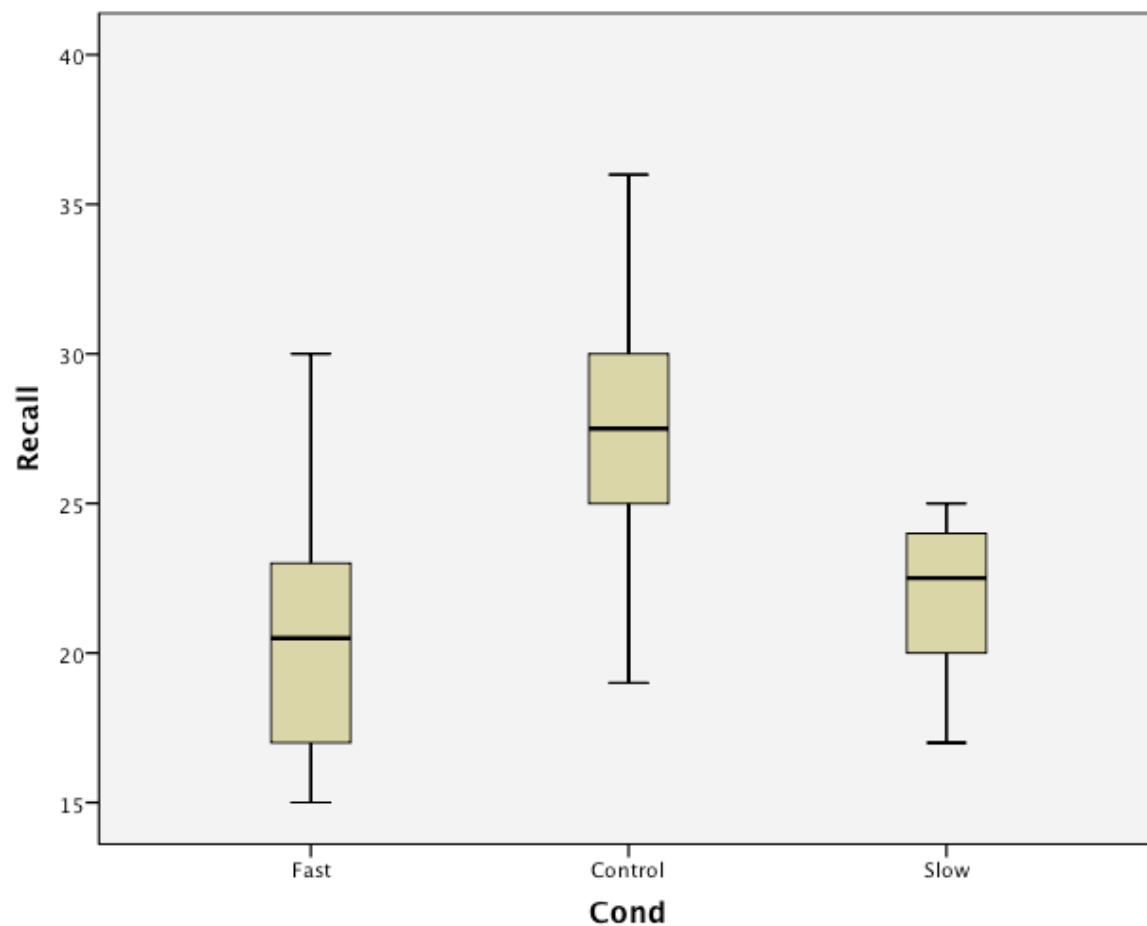
Descriptives

Rule #1 in my class is "always plot your data." I covered this in the first post in the series, so I won't go into too much detail here. Mostly what I want out are boxplots. EXAMINE handles this in SPSS, but first we'll label the levels of the "Cond" variable:

```
ADD VALUE LABELS Cond 0 "Fast" 1 "Control" 2 "Slow".
```

```
EXAMINE VARIABLES= Recall BY Cond
/PLOT BOXPLOT
/COMPARE GROUP
/STATISTICS DESCRIPTIVES.
```

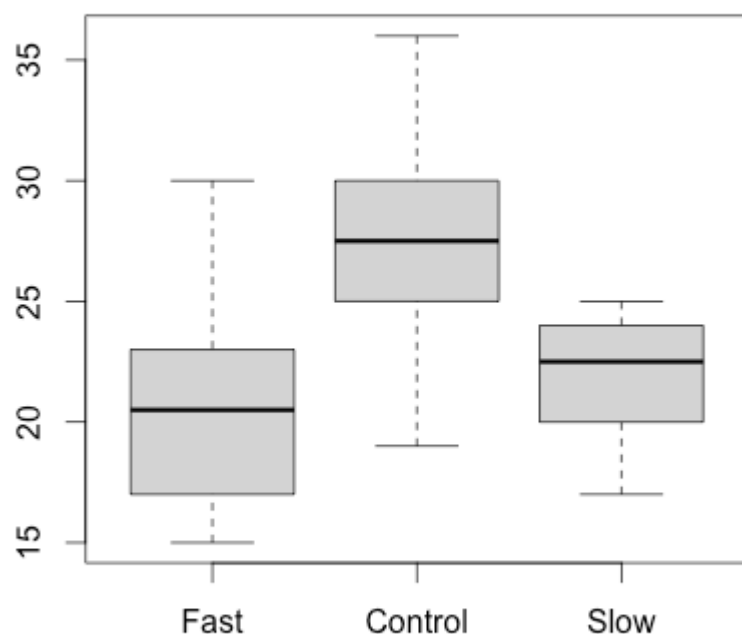
Which produces this output:



Generating the boxplot in R is pretty straightforward. Again, first we'll define the factor with labels and generate a box plot:

```
CondF = factor(Cond, labels = c("Fast", "Control", "Slow"))  
boxplot(Recall ~ CondF, col="lightgray")
```

And here's the output:



Peeve: Again, I don't know what R's deal is with dead whitespace at the top and bottom of the graph.

Regardless, the data look fine and well-behaved, so on with the analysis.

Basic ANOVA

Running the default omnibus ANOVA in SPSS is very easy. There are two choices, ONEWAY and UNIANOVA. As the name suggests, ONEWAY will only handle one-way ANOVA. It's output is a little cleaner than UNIANOVA, but I usually wind up using UNIANOVA because it's more general. I'll show both.

```
ONEWAY Recall BY Cond
/STATISTICS DESCRIPTIVES.
```

Produces this output:

Descriptives									
Recall									
	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum	
					Lower Bound	Upper Bound			
Fast	10	21.20	5.203	1.645	17.48	24.92	15	30	
Control	10	27.40	5.103	1.614	23.75	31.05	19	36	
Slow	10	21.80	2.530	.800	19.99	23.61	17	25	
Total	30	23.47	5.151	.940	21.54	25.39	15	36	

ANOVA

Recall					
	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	233.867	2	116.933	5.895	.008
Within Groups	535.600	27	19.837		
Total	769.467	29			

You can omit the “/STATISTICS DESCRIPTIVES” if you don’t want to see the first table. UNIANOVA gives you one extra option which is nice, the option to print an effect size measure, partial eta squared:

```
UNIANOVA Recall BY Cond
/PRINT ETASQ.
```

Produces this output:

Tests of Between-Subjects Effects

Dependent Variable: Recall						
Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	233.867 ^a	2	116.933	5.895	.008	.304
Intercept	16520.533	1	16520.533	832.813	.000	.969
Cond	233.867	2	116.933	5.895	.008	.304
Error	535.600	27	19.837			
Total	17290.000	30				
Corrected Total	769.467	29				

a. R Squared = .304 (Adjusted R Squared = .252)

We could have asked for descriptives there as well.

As is true of a great many things, R provides many ways to generate a simple oneway ANOVA. Any one of R’s many linear model procedures will work, the most conventional is to just use the “aov()” function and then ask for a table of means and the ANOVA table:

```
Anv = aov(Recall ~ CondF)
model.tables(Anv, "means")
summary(Anv)
```

If you don’t want the table of means, just skip the model.tables() statement. The listed code produces this output:

Tables of means

Grand mean
23.46667

CondF
CondF
Slow Control Fast
21.2 27.4 21.8

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
CondF	2	233.9	116.93	5.895	0.00751 **
Residuals	27	535.6	19.84		

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Peeve: Let me just get this out of the way now: I think the asterisk notation is stupid. It should, at best, be an option, but should not be on by default. People in the R community

frequently act as they are enlightened purists (more on this later), so how can they endorse asterisks? I don't get it.

Peeve: Is it really necessary to report the p -value to five decimal places? I guess it doesn't hurt anything, but still.

Regardless, the default R procedure does not report any effect size measures. Computing partial eta squared by hand isn't difficult, of course, but it's odd that it isn't even an option when it is becoming more and more common that journals at least request, and occasionally require, a metric of effect size. (Incidentally, partial eta squared probably isn't the best one to use, but I'm surprised R reports nothing.) It is possible to get effect size estimates from R using one of the alternative ANOVA procedures; I'll cover that later.

Welch Adjustment

For these particular data the issue of unequal variance isn't all that huge, but this does come up. As I noted with t -tests, the O'Brien's R and Levene tests are slightly easier to do in R; SPSS doesn't even include them as an option. However, sometimes you have unequal variances and it's nice when you can actually handle it. SPSS provides the Welch adjustment as part of the ONEWAY procedure but not UNIANOVA—why is that? The code looks like this:

```
ONEWAY Recall BY Cond
/STATISTICS WELCH.
```

And here's the output:

Robust Tests of Equality of Means				
Recall				
	Statistic ^a	df1	df2	Sig.
Welch	5.072	2	15.914	.020

a. Asymptotically F distributed.

Note that the regular ANOVA table is printed before this.

In R, the base package does include the adjustment in the “oneway.test()” function:

```
oneway.test(Recall ~ CondF)
```

Which yields the following:

One-way analysis of means (not assuming equal variances)

data: Recall and CondF

F = 5.0716, num df = 2.000, denom df = 15.914, p-value = 0.01977

Not much different there.

Contrasts

The omnibus ANOVA null hypothesis is pretty vague; essentially, rejection of the null means “at least one cell mean is different than the grand mean.” That's not very useful in most cases, but far too many people look only at that. I encourage my students to form more specific hypotheses and test those with contrasts. For this particular data set, testing the control vs. the average of the two treatments is sensible—that's (1 -2 1) and testing the two treatments against each other is sensible—that's (1 0 -1). In SPSS both ONEWAY and UNIANOVA support contrasts in a way that's sensible for main effects, though they report slightly different things. Here's ONEWAY:

```

ONEWAY Recall BY Cond
/CONTRAST= 1 -2 1
/CONTRAST=1 0 -1.

```

After the usual ANOVA table, it then produces contrast-relevant output:

Contrast Coefficients						
Contrast	Cond					
	Fast	Control	Slow			
1	1	-2	1			
2	1	0	-1			

Contrast Tests						
		Contrast	Value of Contrast	Std. Error	t	Sig. (2-tailed)
Recall	Assume equal variances	1	-11.80	3.450	-3.420	.002
		2	-.60	1.992	-.301	.766
	Does not assume equal variances	1	-11.80	3.710	-3.181	.006
		2	-.60	1.829	-.328	.748

It's nice that it includes the unequal variance adjustment, but I wish it reported sums of squares. *Peeve*: UNINOVA reports sums of squares, but of course it doesn't report the unequal variance adjustment. Why, SPSS, why? Anyway, here's the UNIANOVA version:

```

UNIANOVA Recall BY Cond
/CONTRAST(cond) = SPECIAL(1 -2 1)
/CONTRAST(cond) = SPECIAL(1 0 -1).

```

The amount of output produced is copious. After the standard table for the omnibus ANOVA, we get these:

Contrast Results (K Matrix)			
			Dependent Variable
Cond	Special Contrast		Recall
L1	Contrast Estimate		-11.800
	Hypothesized Value		0
	Difference (Estimate - Hypothesized)		-11.800
	Std. Error		3.450
	Sig.		.002
	95% Confidence Interval for Difference	Lower Bound	-18.879
		Upper Bound	-4.721

Test Results					
Dependent Variable: Recall					
Source	Sum of Squares	df	Mean Square	F	Sig.
Contrast	232.067	1	232.067	11.699	.002
Error	535.600	27	19.837		

and then:

Contrast Results (K Matrix)

		Dependent Variable
Cond	Special Contrast	Recall
L1	Contrast Estimate	-.600
	Hypothesized Value	0
	Difference (Estimate - Hypothesized)	-.600
	Std. Error	1.992
	Sig.	.766
	95% Confidence Interval for Difference	
	Lower Bound	-4.687
	Upper Bound	3.487

Test Results

Dependent Variable: Recall					
Source	Sum of Squares	df	Mean Square	F	Sig.
Contrast	1.800	1	1.800	.091	.766
Error	535.600	27	19.837		

It's great that it produces the contrast estimates and the confidence intervals, and of course the sum of squares.

Peeve: it never notes in the output what the contrast weights actually were, and there's no way to supply a label. Again, why not include this?

Now, R. There are, of course, many ways to do contrasts in R. The one that takes the least amount of code is the "fit.contrast()" function in the "gmodels" package. Assuming we still have the aov object (named "Anv" above) floating around and the "gmodels" package loaded up, then the two contrasts can be done like this:

```
fit.contrast(Anv, CondF, c(1, -2, 1), df=T)
fit.contrast(Anv, CondF, c(1, 0, -1), df=T)
```

And that produces the following output:

```

              Estimate Std. Error  t value   Pr(>|t|) DF
CondF c=( 1 -2 1 )    -11.8      3.44996 -3.42033 0.002003601 27

              Estimate Std. Error   t value Pr(>|t|) DF
CondF c=( 1 0 -1 )     -0.6      1.991835 -0.3012297 0.765547 27
```

Nice that it has the contrast weight and the contrast estimate in there, but too bad no sums of squares. In general, however, I don't really like the "fit.contrast()" function very much because it doesn't work the way I'd want it to for factorial designs. Probably the more conventional way to handle contrasts in R is to directly set the "contrasts" attribute of the factor variable, and then request that the summary table be split by the individual contrasts. It looks like this:

```
contrasts(CondF) = cbind(c(1, -2, 1), c(1, 0, -1))
CondLab = list("C v T"=1, "Fast v Slow"=2)
AnvL = aov(Recall ~ CondF)
summary(AnvL, split=list(CondF=CondLab))
```

This produces the following output:

```

      Df Sum Sq Mean Sq F value Pr(>F)
CondF    2  233.9   116.93    5.895 0.00751 **
  CondF: C v T    1  232.1   232.07   11.699 0.00200 **
  CondF: Fast v Slow 1    1.8    1.80    0.091 0.76555
Residuals   27  535.6    19.84

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Very nice, though I'd still like to see the contrast estimates and confidence intervals. Also, R only works if you specify orthogonal contrasts. That is, in principle, sensible, but it does mean more work if you want to test a couple contrasts that happen to not be orthogonal.

Posthocs

SPSS supports a huge array of posthocs, many of which are essentially obsolete given more modern techniques, but I don't think one has ever been taken out of SPSS, so many of them are in there. And, thankfully, they're built right in to the ANOVA procedures. That is, both ONEWAY and UNIANOVA include an option for running posthoc procedures. As noted, there are many procedures available such as SNK and Tukey, but personally I'm a fan of the Ryan-Einot-Gabriel-Welsch procedure (also called the "Ryan's Q" procedure):

```

ONEWAY Recall BY Cond
/POSTHOC = QREGW.

```

or

```

UNIANOVA Recall BY Cond
/POSTHOC = Cond(QREGW).

```

The two produce almost identical output after the standard ANOVA table, which is a table showing homogenous subsets:

Recall			
Ryan-Einot-Gabriel-Welsch Range ^a			
Cond	N	Subset	
		1	2
Fast	10	21.20	
Slow	10	21.80	
Control	10		27.40
Sig.		.766	1.000

Means for groups in homogeneous subsets are displayed.

Based on observed means.

The error term is Mean Square(Error) = 19.837.

a. Alpha = .050.

My sense from the R forums and places like StackOverflow is that the R community frowns on posthocs in general. I get that—it's certainly inferior to having good hypotheses in advance and using contrasts—but just because you don't personally like something doesn't mean there aren't still occasions that it can be useful for someone else. I find the paternalistic attitude taken by people in the R community on subjects like this to be incredibly off-putting. Sometimes there are reasons for doing things in ways that are not preferred. We do not live in an idealized statistical world—sometimes the circumstances of real research make other demands.

Anyway, it is that kind of attitude in the R community that I believe is responsible for R being so bad at posthocs. The only posthoc built in to the base package is Tukey's HSD.

Wow, hey, the 1960s called and they'd like their posthoc back. If you're willing to put up with Tukey, well, here's example code and output:

```
TukeyHSD(Anv, "CondF")
```

That uses the `lm` object created by "aov()" earlier. The output is very nice, though I can't imagine why you need the p-value to a whopping *seven* decimal places. Seriously, has anyone ever made a decision that was contingent on that small a probability, ever?

```
Tukey multiple comparisons of means
 95% family-wise confidence level
```

```
Fit: aov(formula = Recall ~ CondF)
```

```
$CondF
      diff      lwr      upr      p adj
Control-Fast  6.2  1.261409 11.1385914 0.0117228
Slow-Fast     0.6 -4.338591  5.5385914 0.9513012
Slow-Control -5.6 -10.538591 -0.6614086 0.0238550
```

It's at least easy to read and produces clear confidence intervals. Too bad it's not as powerful (in the statistical sense of power) as other posthocs.

Now, the Ryan's Q procedure is available in R in the "mutoss" package, but the output is simply awful. Actually, general use of it isn't very smooth, either. The code looks like this:

```
regwq(Recall ~ CondF, data=synchro, alpha=.05)
```

The "data" and "alpha" arguments are mandatory, not optional. It won't look for variable names in the current name space, only in the provided data frame, and it doesn't default to an alpha of .05. Seriously? But wait, it gets better! Here's the output:

```
#---REGWQ - Ryan / Einot and Gabriel / Welsch test procedure
Number of hyp.:      3
Number of rej.:      2
rejected pValues adjPValues
1 3 0.0091 0.0091
2 1 0.0117 0.0117
$adjPValues
[1] 0.0117 0.7655 0.0091
Levels: 0.0091 0.0117 0.7655

$rejected
[1] TRUE FALSE TRUE

$statistic
[1] 4.402 0.426 3.976

$confIntervals
[,1] [,2] [,3]
Control-Slow 6.2 NA NA
Fast-Slow   0.6 NA NA
Control-Fast 5.6 NA NA

$errorControl
An object of class "ErrorControl"
Slot "type":
[1] "FWER"

Slot "alpha":
[1] 0.05
```

```
Slot "k":  
numeric(0)
```

```
Slot "q":  
numeric(0)
```

Warning: Rant ahead. There are so many annoying pieces of this that it's hard to know where to start. First, there's a ton of extraneous crap there. Second, where are the confidence intervals? Third, could there be more indirection here in actually reading the printout? Look, are "Control" and "Slow" different? Turns out they are. The p -values listed under "\$adjPValues" match up with the list of pairs under "\$confIntervals." The first p -value is less than .05, so we can reject, then we can look at the second pair, etc. There's even a "\$rejected" line that just gives the binary hypothesis test result, but again you have to match those up with the pairs yourself. OK, fine, the information you need is actually present, but would it have killed anyone in the R world to actually put the relevant info actually proximal with the list of pairs? This is just ludicrous. It's not really that big a deal with only three pairs, but when you have ten pairs and you care about one of the ones in the middle of the pack, getting it right takes WAY more effort than it should.

This is reflective of a more general problem with R: there are no standards for output quality in R. Like many other open source projects, as long as the person in charge of writing the code for that specific piece can use it, well, that's good enough. In fact, we have to thank the relevant programmer, because if s/he hadn't done it, we'd have nothing. This is one of those cases where the adage "you get what you pay for" is right on the mark. I'm sure some R person will get all offended that I'm criticizing and will say (or at least think) "well, if you're so smart and you can do it so much better, why don't you go ahead and write it?" That kind attitude, which I've seen on R message boards before, completely misses the point. I'm trying to get new graduate students to use this thing. Output like this sends a clear message to them, and that message is "We're going to make simple things unnecessarily hard. Go use SPSS, we don't want you."

Now, for you R fans out there, don't lose heart. There's a fairly major bit of stupidity in SPSS coming up in the next episode (the problem is interaction contrasts). R handles this about a thousand times better than SPSS, so payback is coming.

Advertisements



Gut Doctor "I Beg Americans To Throw Out This Vegetable"

[REPORT THIS AD](#)



Gut Doctor "I Beg Americans To Throw Out This Vegetable"

[REPORT THIS AD](#)

Share this:

Like

Be the first to like this.

Related

[Translating SPSS to R: Factorial Repeated Measures](#)

In "R"

[Translating SPSS to R: Mixed Repeated-Measures ANOVA](#)

In "R"

[Translating SPSS to R: Non-orthogonal \(or unbalanced\) Factorial Between-Subjects ANOVA](#)

In "R"

Categories: [Uncategorized](#) Tags: [R](#), [SPSS](#), [statistics](#)
[Comments \(1\)](#) [Trackbacks \(0\)](#) [Leave a comment](#) [Trackback](#)



1.

[Kristie Campana](#)

2017.10.16 at 09:39

[Reply](#)

Spot on with the criticism. It's too bad the output is so ugly for a procedure I find students need to use all the time to really understand what's going on in their data. Thanks for posting this, it's the best source of info I've been able to find about the Ryan's Q!

1. No trackbacks yet.

Leave a Reply

[Translating SPSS to R: Factorial Between-Subjects ANOVA iPhone 5 and iOS6 Thoughts](#)
[RSS feed](#)

Recent Posts

- [Why My Fantastic New iPhone Annoys Me](#)
- [Reactions to the iPhone7 Announcement](#)
- [Translating SPSS to R: Mixed Repeated-Measures ANOVA](#)
- [My iOS Apps, 2014 Edition](#)
- [My Mac Apps, 2014 Edition](#)

Archives

- [September 2018](#)
- [September 2016](#)
- [August 2015](#)
- [January 2015](#)
- [September 2014](#)
- [July 2014](#)

- [June 2014](#)
- [April 2014](#)
- [February 2014](#)
- [February 2013](#)
- [December 2012](#)
- [November 2012](#)
- [October 2012](#)
- [September 2012](#)

[Twitter Updates](#)

- RT [@dynarski](#): Faculty meetings, explained [twitter.com/netmanchris/st...](#) * [1 hour ago](#)
- RT [@pamelafessler](#): WI finds 24 "suspected" cases of fraud out of 2.7 mil ballots cast last year and that none "would have been prevented by..." * [4 days ago](#)
- RT [@emdoyl](#): 2011: The world's gonna end next year...like probably....bc of something w the Mayans People: *freaking out* 2019: There is S... * [2 weeks ago](#)

Follow [@MikeByrneHFHCl](#)

October 2012

M T W T F S S

1 [2](#) [3](#) 4 5 6 7

[8](#) 9 [10](#) 11 [12](#) 13 14

15 16 17 18 [19](#) 20 21

22 23 24 25 26 27 28

29 30 31

[« Sep](#) [Nov »](#)

Meta

- [Register](#)
- [Log in](#)
- [Entries RSS](#)
- [Comments RSS](#)
- [WordPress.com](#)

Advertisements



**Gut Doctor "I Beg Americans
To Throw Out This
Vegetable"**

REPORT THIS AD

[Top](#)
[Blog at WordPress.com.](#)
