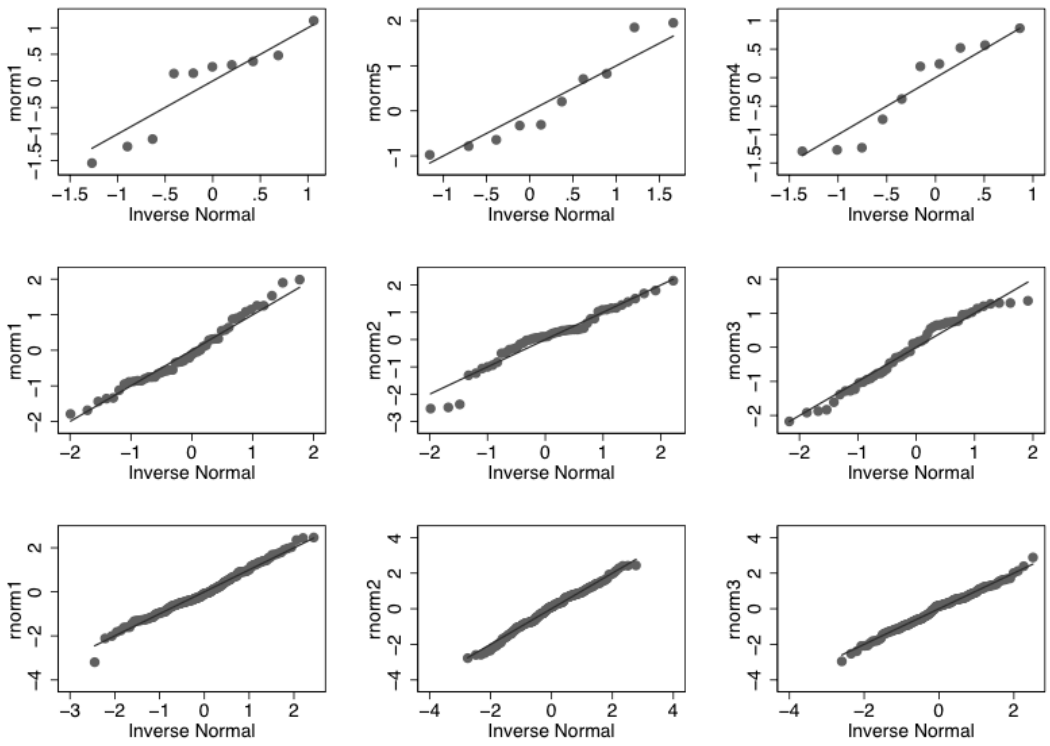
Biostatistics 208 HW #1, Solutions

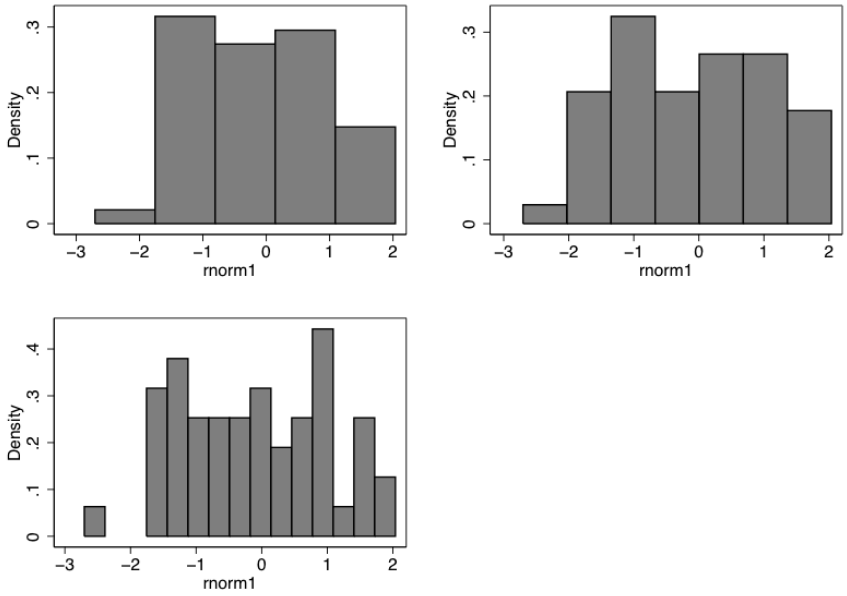
Problem 2.2

Normal Q-Q plots are shown below for multiple samples of size 10, 50 and 200, all randomly generated from a Normal distribution. With samples of size 10, the data points can diverge rather badly from the diagonal straight line. In contrast, with samples of size 50 and especially 200, the plots are pretty reassuring. Even there, some noise in the tails is to be expected.



Problem 2.3

The histograms for the sample I drew do not look particularly Normal with 5, 7, or 15 bins; they suggest a short-tailed, broad-shouldered distribution. As we know from the previous problem, this is consistent in small samples with data from a Normal distribution; and as compared to long tails, short tails cause fewer problems for statistical procedures built on the Normality assumption. The coarseness of the histogram shows that it is less useful for assessing Normality than the more specific Normal Q-Q plot. With this sample size, 7 bins (the Stata default) is probably best, providing better resolution than 5 bins but less noise than 15.



Problem 2.4

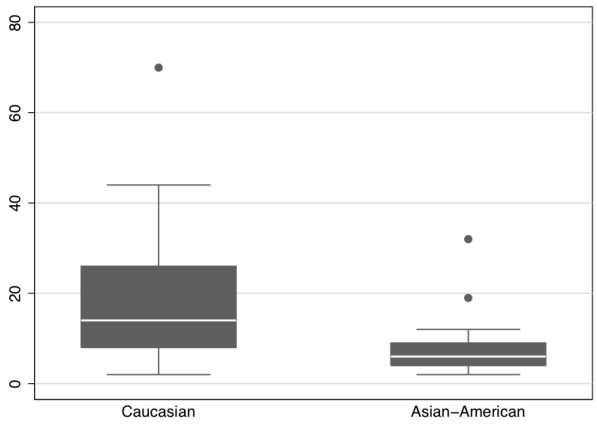
1. At the outset, it is a good idea to check the mean and SD as well as the more robust 5-number summary, in case the data are badly skewed. The results tabulated below were obtained using the command

```
bysort group: sum times, detail
```

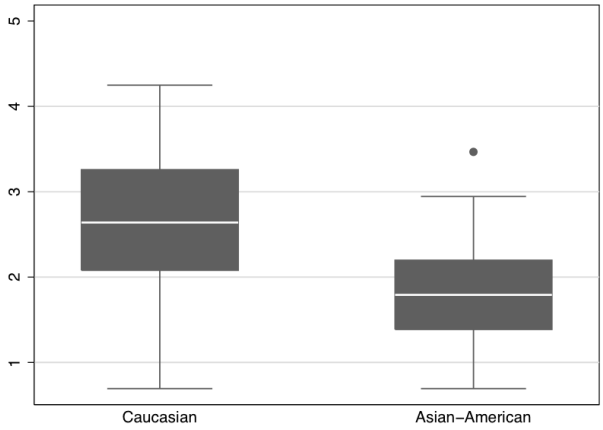
Note that there are 19 observations for Caucasians, an error we caught after the first printing of the textbook.

	Caucasian	**Asian American**
minimum	2	2
25% pctile	7	4
median	14	6
75% pctile	26	9
maximum	70	32
mean	19	8
SD	17	7

2. Side-by-side boxplots are most appropriate for comparing calibration times across the two groups. (Overlaid density plots would also be OK for this.)



3. Both the numerical summary and the boxplots above show that the calibration times are longer and more variable for Caucasians. There are one or two outliers on the right in both groups.



4. The boxplots above show that log transformation "pulls in" the outlying calibration times, reduces right skewness, and makes the variances more similar. In many contexts, the variance of the data is roughly proportional to the conditional mean – in this case the mean for each group. When that is true, a log transformation will equalize variances. Log transformation will be especially helpful in regression. While the medians also appear closer in the boxplots, keep in mind that the differences are now measured on the (natural) log scale.

5. One should consider adjusting for any variables that differ across the groups being compared and are likely to affect the calibration time (i.e., potential confounders).

Problem 2.5

1. The same numerical summary can be used as in the previous problem.

	IVT+	**IVT-**
Minimum	70	40
25% pctile	96	85
median	117	98
75% pctile	142	120
maximum	182	180
mean	120	104
SD	31	31

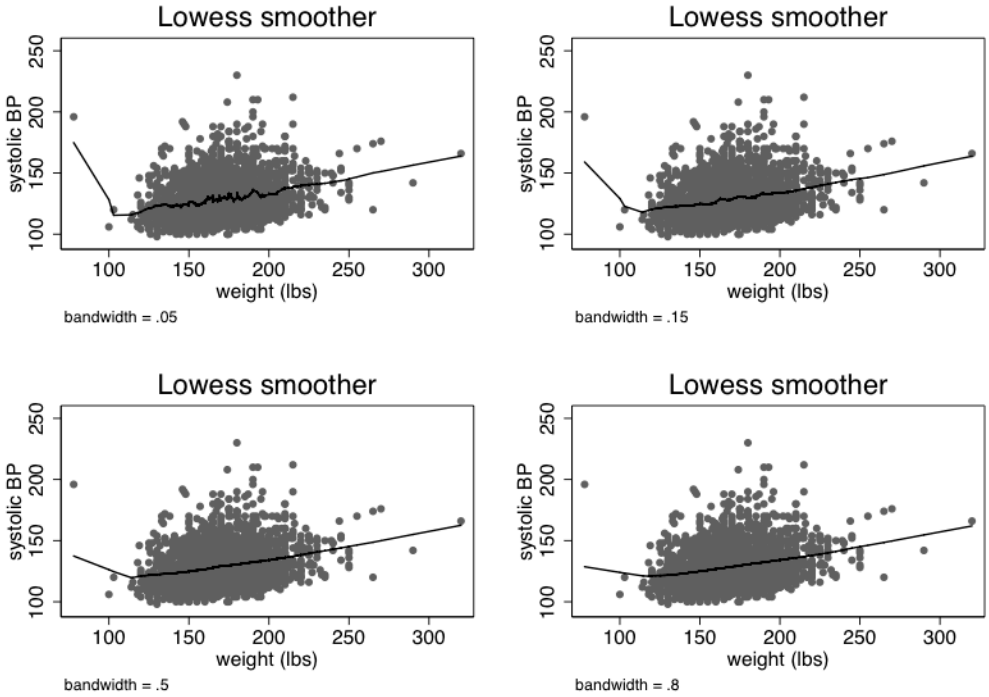
We see that both mean and median QRS times are longer in the IVT+ group. There is at most a little evidence of skewness (means only slightly larger than medians within groups) and the SDs are the same across groups.

2. Side-by-side boxplots are again appropriate. (Overlaid density plots would also be OK for this.) They show that QRS times are about 20 msec longer in the IVT+ compared with the IVT- patients. The spread of the data is quite similar in the two groups.

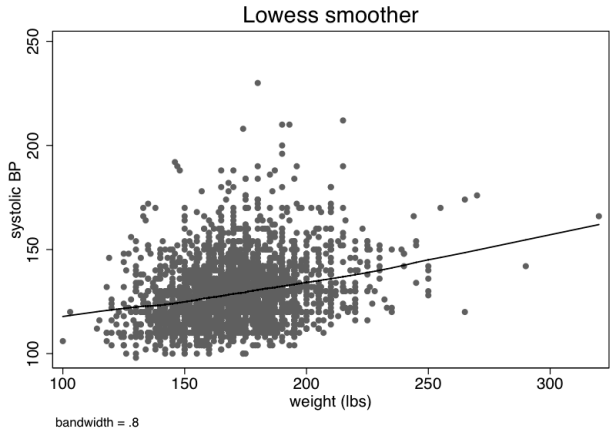
3. To examine QRS time classified as normal vs. abnormal, we would define a binary variable (say `qrs_abn`) equal to one for subjects with QRS times > 120 msec and equal to zero for those with times ≤ 120 msec. Then we would cross-tabulate this outcome with IVT status. The table below shows that only 29% of patients with normal QRS times have IVT, while 47% of pts. with abnormal QRS times have it.

```
. tab qrs_abn ivt, row;

+----------------+
| Key            |
|----------------|
|   frequency    |
| row percentage |
+----------------+

           | Inducible ventricular
QRS time > |      tachycardia
  120 msec |  negative   positive |     Total
-----------+----------------------+----------
        no |        27         11 |        38
           |     71.05      28.95 |    100.00
-----------+----------------------+----------
       yes |         8          7 |        15
           |     53.33      46.67 |    100.00
-----------+----------------------+----------
     Total |        35         18 |        53
           |     66.04      33.96 |    100.00
```

Problem 2.6
The bandwidths are shown below each Lowess plot.



All four plots are *J*-shaped, but show essentially linear increase to the right of the minimum fitted SBP, at about 120 pounds. The higher bandwidths filter out the jagged appearance and reduce the upward tilt on the left, which is entirely due to a single observation. Dropping that influential point gets rid of the *J*, as shown in the Lowess plot below. Although this is a large dataset, so that Lowess plots with narrower bandwidths are not all that noisy, in this instance little is gained by using bandwidths narrower than the default value of 0.8.



Part 2

1. The `tab, sum()` command shows mean SBP, as well as the sample size and SD, in each of the four behavioral pattern groups.

```
. tab behpat, sum(sbp)

 behavioral |
 pattern (4 |       Summary of systolic BP
     level) |        Mean   Std. Dev.       Freq.
------------+------------------------------------
         A1 |   129.24621    15.29221         264
         A2 |   129.88906   15.770845        1325
         B3 |    127.5551   14.787947        1216
         B4 |   127.15473   13.101246         349
------------+------------------------------------
      Total |   128.63285   15.117731        3154
```

2. The command and regression results are as follows:

```
. reg sbp i.behpat

      Source |       SS       df       MS              Number of obs =    3154
-------------+------------------------------           F(  3,  3150) =    6.40
       Model |  4365.19641     3  1455.06547           Prob > F      =  0.0003
    Residual |  716239.641  3150  227.377664           R-squared     =  0.0061
-------------+------------------------------           Adj R-squared =  0.0051
       Total |  720604.837  3153  228.545778           Root MSE      =  15.079

------------------------------------------------------------------------------
         sbp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      behpat |
          2  |   .6428445   1.016309     0.63   0.527    -1.349851     2.63554
          3  |  -1.691113   1.023849    -1.65   0.099    -3.698592    .3163655
          4  |  -2.091484   1.229956    -1.70   0.089     -4.50308    .3201111
             |
       _cons |   129.2462   .9280512   139.27   0.000     127.4266    131.0659
------------------------------------------------------------------------------
```

3. The values of the four indicators are as follows. As in the examples and lab using the HERS categorical variable `physact`, we only need three of the four to distinguish the four groups; the indicator for the group with the lowest numeric code (A1) is automatically omitted by Stata from the regression model.

Behavioral pattern	**1**	**2**	**3**	**4**
A1	1	0	0	0
A2	0	1	0	0
B3	0	0	1	0
B4	0	0	0	1

4. I included the computation for group A1, although it could be read directly from the regression output (the coefficient for `_cons`). These results match the output from the `tab, sum()` command (with tiny discrepancies due to different rounding conventions), though this would not be true if there were covariates in the model.

```
. * group A1;
. lincom _cons
 ( 1)  _cons = 0
------------------------------------------------------------------------------
         sbp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   129.2462   .9280512   139.27   0.000     127.4266    131.0659
------------------------------------------------------------------------------
. * group A2;
. lincom _cons + 2.behpat
 ( 1)  2.behpat + _cons = 0
------------------------------------------------------------------------------
         sbp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   129.8891   .4142533   313.55   0.000     129.0768    130.7013
------------------------------------------------------------------------------
. * group B3;
. lincom _cons + 3.behpat
 ( 1)  3.behpat + _cons = 0
------------------------------------------------------------------------------
         sbp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   127.5551   .4324213   294.98   0.000     126.7072     128.403
------------------------------------------------------------------------------
. * group B4;
. lincom _cons + 4.behpat
 ( 1)  4.behpat + _cons = 0
------------------------------------------------------------------------------
         sbp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   127.1547   .8071629   157.53   0.000     125.5721    128.7373
------------------------------------------------------------------------------
```

As we saw in lab, this can also be done more simply using the `margins` command:

```
. margins behpat
Adjusted predictions                            Number of obs   =       3154
Model VCE    : OLS

Expression   : Linear prediction, predict()
------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      behpat |
           1 |   129.2462   .9280512   139.27   0.000     127.4273    131.0652
           2 |   129.8891   .4142533   313.55   0.000     129.0771     130.701
           3 |   127.5551   .4324213   294.98   0.000     126.7076    128.4026
           4 |   127.1547   .8071629   157.53   0.000     125.5727    128.7367
------------------------------------------------------------------------------
```

5. The `testparm` result shown below confirms that there is statistically significant heterogeneity in mean SBP across the four behavioral pattern groups. Again, because this model includes no covariates, the model *F*-test shown in the `regress` output agrees exactly with this output. Recall that this test is sensitive to any departure from homogeneity across groups.

```
. testparm i.behpat
 ( 1)  2.behpat = 0
 ( 2)  3.behpat = 0
 ( 3)  4.behpat = 0
       F(  3,  3150) =    6.40
            Prob > F =    0.0003
```

6. The output for the fitted model is below:

```
. reg sbp ibn.behpat, hascons

      Source |       SS           df       MS      Number of obs   =     3,154
-------------+----------------------------------   F(3, 3150)      =      6.40
       Model |  4365.19641         3  1455.06547   Prob > F        =    0.0003
    Residual |  716239.641     3,150  227.377664   R-squared       =    0.0061
-------------+----------------------------------   Adj R-squared   =    0.0051
       Total |  720604.837     3,153  228.545778   Root MSE        =    15.079

------------------------------------------------------------------------------
         sbp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      behpat |
          A1 |   129.2462   .9280512   139.27   0.000     127.4266    131.0659
          A2 |   129.8891   .4142533   313.55   0.000     129.0768    130.7013
          B3 |   127.5551   .4324213   294.98   0.000     126.7072     128.403
          B4 |   127.1547   .8071629   157.53   0.000     125.5721    128.7373
------------------------------------------------------------------------------
```

The predictors in the model without an intercept are just the four indicators for each behavior group. So, the coefficients in this case represent actual group mean values rather than differences between level-specific group means and the mean for the reference level. Therefore, although the homogeneity null hypothesis for this model is still that the true group means are all equal, applying `testparm` as before doesn't lead to the same result:

```
. testparm i.behpat

 ( 1)  1bn.behpat = 0
 ( 2)  2.behpat = 0
 ( 3)  3.behpat = 0
 ( 4)  4.behpat = 0

       F(  4,  3150) =57384.42
            Prob > F =    0.0000
```

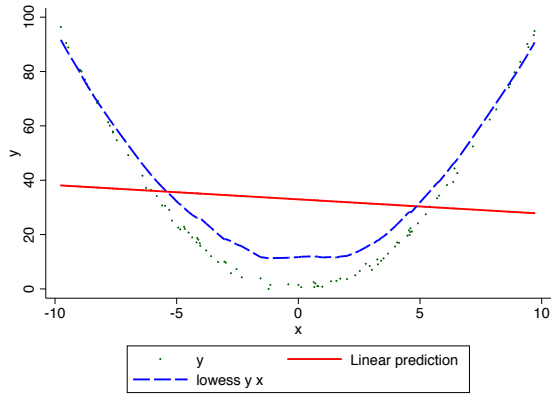
This command is clearly still evaluating the null hypothesis that the group means are zero – not what we want. The trick is to specify the `equal` option as follows:

```
. testparm i.behpat, equal

 ( 1)  - 1bn.behpat + 2.behpat = 0
 ( 2)  - 1bn.behpat + 3.behpat = 0
 ( 3)  - 1bn.behpat + 4.behpat = 0

       F(  3,  3150) =    6.40
            Prob > F =    0.0003
```

Optional problems

O1. The Stata output below generates 100 values of a random variable x with a uniform[-10, 10] distribution, a random error r (distributed as normal with mean=0 & SD=1), and an outcome y, represented as the sum of the assumed component ($E[y|x] = x^2$), and the error term. The Pearson correlation between y and x is then estimated. The resulting correlation is approximately -0.1. A scatter plot, linear regression fit and lowess smooth of the empirical relationship are also presented. The results suggest that lack of an observed correlation (or, equivalently lack of a linear relationship in a simple regression) can be misleading. In this case, the data display a clear nonlinear (quadratic) relationship, but this is missed by the standard summaries that look only for a linear relationship.

```
. set obs 100
number of observations (_N) was 0, now 100

. gen x = runiform(-10,10)

. gen r = rnormal(0,1)

. gen y = x^2 + r

. corr y x
(obs=100)

             |        y        x
-------------+------------------
           y |   1.0000
           x |  -0.0889   1.0000
. regress y x

      Source |       SS           df       MS      Number of obs   =       100
-------------+----------------------------------   F(1, 98)        =      1.06
       Model |  905.172365         1  905.172365   Prob > F        =    0.3063
    Residual |  83889.6116        98  856.016444   R-squared       =    0.0107
-------------+----------------------------------   Adj R-squared   =    0.0006
       Total |  84794.7839        99  856.512969   Root MSE        =    29.258

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |  -.5246367   .5101925    -1.03   0.306    -1.537097    .4878237
       _cons |   32.96414   2.934459    11.23   0.000      27.1408    38.78747
------------------------------------------------------------------------------

. predict fitted, xb

. twoway (scatter y x, msize(vtiny)) (line fitted x, sort lpattern(solid) lcolor(red)
lwidth(medthick)) (lowess y x, lpattern(longdash) lcolor(blue) lwidth(medthick)),
plotregion(style(none)) scheme(s1color) ytitle ("y") xtitle("x")
```



O2. The Stata code below provides a simulation program `swpw` that generates a gamma distributed random variable with specified shape and scale parameters (supplied as arguments `shp` and `scl`), and using a chosen sample size `n`, runs the Shapiro-Wilk test and uses the resulting *p*-value to define a scalar indicator `rej` of a rejection at the 0.05 level.

```
program define swpw, rclass
args n shp scl
drop _all
set obs `n'
* generate a gamma random variable with chosen shape and scale parameters
tempvar y
gen `y' = rgamma(`shp', `scl')
* test for normality
swilk `y'
* extract p-value, and classify
tempname rej
if r(p) < 0.05 {
scalar `rej' = 1
}
if r(p) >= 0.05 {
scalar `rej' = 0
}
return scalar rej = `rej'
end
```

The output below shows results from running the program for the specified shape and scale parameter choices, and the desired sample sizes using the `simulate` command with 5000 repetitions. For each simulation, the `summarize` command is applied to a generated binary indicator (`rej`) to estimate the empirical proportion of rejections. The latter provides an empirical estimate of power.

O2. Stata output

The following commands run the simulation program with 5000 repetitions for the desired choices for sample size, shape and scale. Note that using the random seed (12619) provided, the simulation results will be identical to the results tabulated below.

```
set seed 12619
simulate pval=r(rej), reps(5000) nodots: swpw 25 2.5 4
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 50 2.5 4
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 100 2.5 4
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 25 5 2
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 50 5 2
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 100 5 2
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 25 9 1.1
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 50 9 1.1
summarize
simulate pval=r(rej), reps(5000) nodots: swpw 100 9 1.1
summarize
```

Below, I've edited the output from the resulting `summarize` commands into a table giving the results:

	Reps	Power	n	shp	scl
pval	5,000	.553	25	2.5	4
pval	5,000	.8884	50	2.5	4
pval	5,000	.998	100	2.5	4
pval	5,000	.3112	25	5	2
pval	5,000	.5816	50	5	2
pval	5,000	.907	100	5	2
pval	5,000	.1866	25	9	1.1
pval	5,000	.369	50	9	1.1
pval	5,000	.6564	100	9	1.1

The results that power increases with sample size for all choices of shape and scale, and also that power declines with increasing symmetry for non-normal (gamma) distributions.

O3. Based on the hint provided, the sum of squares can be factored as follows:

$$\sum_{i=1}^{n}(y_i - \bar{y})^2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 + 2\sum_{i=1}^{n}(y_i - \hat{y}_i)(\hat{y}_i - \bar{y})$$

Referring to the definitions, this states that TSS = MSS + RSS + *extra term*. For the identity to be true, we have to show that the *extra term* is zero. The algebra below expands this term into pieces: $\sum_{i=1}^{n}(y_i - \hat{y}_i)(\hat{y}_i - \bar{y}) = \sum_{i=1}^{n}(y_i - \hat{y}_i)\hat{y}_i + \sum_{i=1}^{n}(y_i - \hat{y}_i)\bar{y}$

The last term on the right above has to equal zero since the mean is constant and the sum of the residuals is defined to be zero. The leaves us to show that the remaining term is zero as well. Setting $\varepsilon_i = y_i - \hat{y}_i$ and recognizing that $\hat{y}_i = \beta_0 + \beta_1 x_i$

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)\hat{y}_i = \sum_{i=1}^{n}\varepsilon_i\ (\beta_0 + \beta_1 x_i) = \beta_0 \sum_{i=1}^{n}\varepsilon_i + \beta_1 \sum_{i=1}^{n}\varepsilon_i\ x_i$$

We already know that the residuals sum to zero, so we're done if we can show that last term (a weighted sum of residuals with weights x_i) is zero as well. Using again the definition of the residuals, we can rewrite this

$$\sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_i))\ x_i = \sum_{i=1}^{n} y_i x_i - \beta_0 \sum_{i=1}^{n} x_i - \beta_1 \sum_{i=1}^{n} {x_i}^2 \qquad (*)$$

The key to seeing that this last equation equals zero is to refer to the following formula for the least squares solution for β_1 (see slide 29 of Lecture 1):

$$\frac{d}{d\beta_1}\sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_i))^2 = 0 \qquad (**)$$

Taking the derivative, this can be rewritten

$-2(\sum_{i=1}^{n} y_i x_i - \beta_0 \sum_{i=1}^{n} x_i - \beta_1 \sum_{i=1}^{n} {x_i}^2) = 0$, which implies that

$$\sum_{i=1}^{n} y_i x_i - \beta_0 \sum_{i=1}^{n} x_i - \beta_1 \sum_{i=1}^{n} {x_i}^2 = 0$$

From this last equality, we see that the expression (*) above must equal zero, because the least square estimates of the coefficients β_0 and β_1 by definition solve equation (**). Note that there are other ways to go about proving this.