Formula sheet

You are not expected to memorize how to compute these quantities. Each is given in words and in symbols. Do not assume that any given question will need one. When one does, deciding which to use, what to put into it, and what to do with the result is up to you.

One variable	How it is computed	In symbols
Mean	Add up all the values and divide by the number of values.	$\overline{Y} = \frac{\sum_{i=1}^{n} Y_i}{n}$
Median	Sort the values and take the middle one (with an even count, average the two middle ones).	middle value of the sorted Y_i
Range	The maximum minus the minimum.	$\max(Y) - \min(Y)$
Interquartile range (IQR)	Q3 minus Q1.	$\text{IQR} = Q3 - Q1$
Outlier cutoffs	Lower: Q1 minus 1.5 times the IQR. Upper: Q3 plus 1.5 times the IQR.	lower: $Q1 - 1.5 \times \text{IQR}$ upper: $Q3 + 1.5 \times \text{IQR}$
Relative frequency	The count for a level divided by the total number of values.	$\frac{\text{count}}{n}$
Deviation	A value minus the mean.	$Y_i - \overline{Y}$
Mean absolute deviation (MAD)	Add up the absolute value of each value minus the mean (ignore the sign), then divide by the number of values. (SAD is the sum before dividing.)	$\text{MAD} = \frac{\sum_{i=1}^{n} \lvert Y_i - \overline{Y} \rvert}{n}$ $\text{SAD} = \sum_{i=1}^{n} \lvert Y_i - \overline{Y} \rvert$
SS Total	Add up the squared difference between each value and the mean.	$SS_{\text{Total}} = \sum_{i=1}^{n} (Y_i - \overline{Y})^2$
Variance	SS Total divided by (the number of values minus 1).	$s^2 = \frac{SS_{\text{Total}}}{n-1}$
Standard deviation	The square root of the variance.	$s = \sqrt{s^2} = \sqrt{\frac{\sum_{i=1}^{n} (Y_i - \overline{Y})^2}{n-1}}$
SS Total from a variance	The variance times (the number of values minus 1).	$SS_{\text{Total}} = s^2(n-1)$

z-scores	How it is computed	In symbols
z-score	(A value minus the mean) divided by the standard deviation.	$z = \frac{Y - \overline{Y}}{s}$
Empirical rule	About 68%, 95% and 99.7% of values are within 1, 2 and 3 standard deviations of the mean. About 16% are in each tail beyond 1, and 2.5% beyond 2.	$\approx 68\%$ within $\overline{Y} \pm 1s$ $\approx 95\%$ within $\overline{Y} \pm 2s$ $\approx 99.7\%$ within $\overline{Y} \pm 3s$

Models and fit	How it is computed	In symbols
Residual	The actual value minus the value the model predicts for it.	$e_i = Y_i - \widehat{Y}_i$
SS Error	Add up the squared difference between each value and the model's predicted value.	$SS_{\text{Error}} = \sum_{i=1}^{n} (Y_i - \widehat{Y}_i)^2$
SS Model	SS Total minus SS Error.	$SS_{\text{Model}} = SS_{\text{Total}} - SS_{\text{Error}}$
PRE (proportional reduction in error)	SS Model divided by SS Total. Equivalent: (SS Total minus SS Error) divided by SS Total.	$PRE = \frac{SS_{\text{Model}}}{SS_{\text{Total}}} = \frac{SS_{\text{Total}} - SS_{\text{Error}}}{SS_{\text{Total}}}$

Models and fit (continued)	How it is computed	In symbols
Degrees of freedom (df)	df Total: number of values minus 1. df Model: number of parameters minus 1 (groups minus 1 for a group model; 1 for a regression with one explanatory variable). df Error: df Total minus df Model.	$\text{df}_{\text{Total}} = n - 1$ $\text{df}_{\text{Model}} = k - 1$ $\text{df}_{\text{Error}} = \text{df}_{\text{Total}} - \text{df}_{\text{Model}} = n - k$ (k = number of parameters)
MS (error per degree of freedom)	A row's SS divided by its df. Do this for Model, Error and Total.	$MS = \frac{SS}{\text{df}}$
F ratio	MS Model divided by MS Error. Equivalent: (PRE divided by df Model) divided by ((1 minus PRE) divided by df Error).	$F = \frac{MS_{\text{Model}}}{MS_{\text{Error}}} = \frac{PRE/\text{df}_{\text{Model}}}{(1-PRE)/\text{df}_{\text{Error}}}$
Typical residual	The square root of MS Error.	$\sqrt{MS_{\text{Error}}}$
Correlation (r)	The slope of the regression line when both variables are z-scores. Equivalent: add up the products of each pair of z-scores and divide by (the number of pairs minus 1).	r = slope of z_y predicted from z_x $r = \frac{\sum z_x z_y}{n-1}$
r squared	The correlation times itself. With one explanatory variable, this equals PRE.	r^2 (= PRE with one explanatory variable)
Cohen's d	(Mean of group 1 minus mean of group 2) divided by the pooled standard deviation. Pooled standard deviation: the square root of the average of the two groups' variances, each weighted by its df (number of values minus 1); for a group model, the square root of MS Error.	$d = \frac{\overline{Y}_1 - \overline{Y}_2}{s_{\text{pooled}}}$ $s_{\text{pooled}} = \sqrt{\frac{\text{df}_1 s_1^2 + \text{df}_2 s_2^2}{\text{df}_1 + \text{df}_2}} = \sqrt{MS_{\text{Error}}}$ for a group model

ANOVA table layout

	SS	df	MS	F	PRE
Model	SS_{Model}	df_{Model}	MS_{Model}	$\frac{MS_{\text{Model}}}{MS_{\text{Error}}}$	$\frac{SS_{\text{Model}}}{SS_{\text{Total}}}$
Error	SS_{Error}	df_{Error}	MS_{Error}		
Total	SS_{Total}	df_{Total}			

Blank cells are not shown. In R, supernova() adds a p-value column.

Inference	How it is computed	In symbols
F and t (two groups)	F equals the t value squared.	$F = t^2$
95% confidence interval	The estimate plus and minus 2 times its standard error. (The 2 applies to 95% only.)	$\text{estimate} \pm 2 \times SE$
Bootstrap 95% bounds	Sort the 1000 bootstrapped b_1s. The bounds are the 26th and 975th values.	26th and 975th of the 1000 sorted bootstrapped b_1s

R commands

You will not be asked to write R code, and reading R code or output is not guaranteed to appear in any question. These commands are here in case it does. Y is the outcome variable, X is the explanatory variable, and data_set is the data frame.

Commands	What they do
Base R	
`x <- 5` `c(1, 2, 3)` `x[3]` `sum()` `print()`	Assign; combine values; third element; add; print
`==` `!=` `>` `<` `>=` `<=` `NA`	Comparisons; a missing value
Data	
`str(data_set)` `head(data_set)` `tail(data_set)` `data_set$Y`	Structure; first six rows; last six rows; one column
`select(data_set, Y1, Y2)` `filter(data_set, Y > 300)` `arrange(data_set, Y)`	Keep variables; keep rows that meet a condition; sort rows
`na.omit(data_set)` `is.na(Y)`	Drop rows with missing values; test for missing
`mutate()` `recode()` `factor(data_set$Y)` `as.numeric(data_set$Y)`	Create a variable; recode values; make categorical; make numeric
Summaries	
`favstats(~ Y, data = data_set)` `favstats(Y ~ X, data = data_set)`	Min, Q1, median, Q3, max, mean, sd, n, missing; the same by group
`tally(~ Y, data = data_set)` `tally(Y ~ X, data = data_set, margins = TRUE, format = "proportion")`	Frequency table; two-way frequency table
`mean()` `median()` `var()` `sd()` `sqrt()`	Mean, median, variance, standard deviation, square root
`cor(Y ~ X, data = data_set)` `cohensD(Y ~ X, data = data_set)`	Correlation; Cohen's d
`b0()` `b1()` `pre()` `f()`, as in `b1(Y ~ X, data = data_set)`	Just b_0, b_1, PRE or F (each takes a formula or a fitted model)
Graphs	
`gf_histogram(~ Y, data = data_set, bins = , binwidth = )` `gf_dhistogram()`	Histogram (counts); histogram (density)
`gf_boxplot(~ Y)` `gf_bar(~ Y)` `gf_props()` `gf_percents()`	Box plot; bar graph; proportions; percents
`gf_point(Y ~ X)` `gf_jitter(Y ~ X, width = , height = 0)` `gf_boxplot(Y ~ X)`	Scatterplot; jittered scatterplot; box plots by group
`gf_facet_grid(X ~ .)` `gf_model(model)` `gf_lm()`	Panels by group; add model predictions; add a regression line
`gf_vline(xintercept = )` `gf_hline(yintercept = )`	Add a vertical or horizontal line
Models	
`empty_model <- lm(Y ~ NULL, data = data_set)`	Fit the empty model (the intercept is the mean)
`my_model <- lm(Y ~ X, data = data_set)`	Fit a model with one explanatory variable
`predict(my_model)` `resid(my_model)` `sum(resid(my_model)^2)`	Predictions; residuals; SS Error
`supernova(my_model)`	ANOVA table (columns SS, df, MS, F, PRE, p)
Probability	
`zscore(data_set$Y)` `xpnorm(65.1, mean = , sd = )`	Z-scores; area under a normal curve
`xpf(sample_f, df1 = 2, df2 = 10)`	p-value from the F-distribution
Simulation	
`shuffle(data_set$Y)` `resample(data_set)` `sample(x, n, replace = TRUE)` `set.seed()`	Mix up values; sample rows with replacement; sample with replacement; fix the random seed
`sdob1 <- do(1000) * b1(shuffle(Y) ~ X, data = data_set)`	1000 b_1s from shuffled data (centered on 0)
`sdob1_boot <- do(1000) * b1(Y ~ X, data = resample(data_set))`	1000 bootstrapped b_1s (centered on the sample b_1)
`gf_histogram(~ b1, data = sdob1, fill = ~middle(b1, .95))` `lower()` `upper()`	Shade the middle 95% of a sampling distribution; the lower and upper values
`tally(~ b1 > sample_b1, data = sdob1, format = "proportion")` `tally(sdob1$b1 > sample_b1 \| sdob1$b1 < -sample_b1)`	Proportion of b_1s above the sample b_1 (one tail); count of b_1s at either extreme
`sd(~ b1, data = sdob1)`	Standard deviation of the b_1s (the standard error)
Intervals and tests	
`confint(my_model)` `confint(my_model, level = .99)`	Confidence intervals (columns 2.5% and 97.5%; the intercept row is β_0, the X row is β_1)
`summary(my_model)$coef`	Estimates, standard errors, t values and p-values
`t.test(Y ~ X, data = data_set, var.equal = TRUE)`	t-test using pooled variance
`pairwise(my_model, correction = "Bonferroni")` `plot = TRUE`	Pairwise comparisons (corrections: "Tukey", "Bonferroni", "none")
`xchisq.test(Y ~ X, data = data_set)`	Chi-square test