Stat 20 Midterm Cheat Sheet

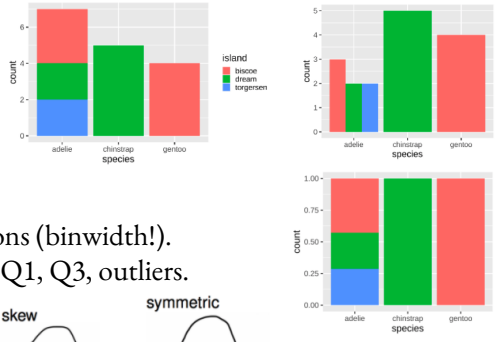
R Data Transformations

Summarize → Compute table of summaries. **Ex**: summarize(penguins, avg = mean(weight))
→ **Functions**: n, n_distinct, sum(!is_na()), mean, median, first, last, nth, quantile, min, max, iqr, mad, sd, var
Count → Count number of rows in each group defined by the variables in ... Also **tally**(). **Ex**: count(penguins, color)
Group_by → create a "grouped" copy of a table grouped by columns in dplyr functions will manipulate each "group" separately and combine the results. **Ex**: penguins %>% group_by(color) %>% summarize(avg = mean(weight))
Distinct → Remove rows with duplicate values
Slice → Select rows by position. **Ex**: slice(penguins, 10:15)
Arrange → Order rows by values of a column or columns, **desc**() for high to low. **Ex**: arrange(penguins, desc(weight))
Add_row → Add one or more rows to the table. Ex: add_row(penguins, weight=1, color="blue")
Select → Extract columns as a table. Ex: select(penguins, weight, height)
Across → Summarize or mutate multiple columns in the same way. Ex: summarize(penguins, across(everything(), mean))
Mutate → Compute new columns, also **add_column()**, **add_count**(), **add_tally**(). **Ex**: mutate(penguins, cute = weight*10)
Logical/Boolean Operators with Filter → ==, < > <= >=, |, !, &, xor(), is.na(), !is.na(), %in%
**Taking the mean of a logical vector/column finds the proportion of rows that are TRUE.

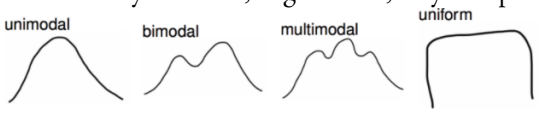
Basics of Data

Four Types: Descriptive, Generalization, Causal, Prediction
Taxonomy of Data: Numerical → Continuous (height), Discrete (size of household); Categorical → Ordinal (ranking), Nominal (name)
Contingency Table: summary table of counts across the combination of levels in two or more variables.
Response Variable: the variable of primary interest
Explanatory Variable: the variable used to explain the response variable.

Visualization



Stacked bar charts: Ased to visualize counts across two variables, useful for overall counts.
Side-by-side bar charts (aka "dodged"): Used to visualize counts across two variables.
Normalized stacked bar charts: Used to visualize condition proportions.
Dot/Scatter plot: all information must be preserved.
Histogram: balance general trends in the distribution with fidelity to the individual observations (binwidth!).
Boxplot: reveal summary statistics, flag outliers, easy comparison between groups, see median, Q1, Q3, outliers.
Modality:



Skewness:

Measures of Center and Spread

Mean: synthesize the magnitudes, good default for symmetric data, sensitive to outliers/small/large values
Median: select a single typical value from the middle, good when data is skewed left or right
Mean: most common value, only option for categorical data
Range: max−min, very sensitive to extreme values
Inner Quartile Range (IQR): median of the larger half (Q3) – median of smaller half (Q1), resistant to outliers, boxplot width
Mean Absolute Deviation (MAD): resistant to outliers $\frac{1}{n}\sum_{i=1}^{n}|x_i-\bar{x}|$
Sample Variance: moderately sensitive to outliers $s^2: \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2$
Sample SD: moderately sensitive to outliers, measured in units of original data $s: \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2}$
Ggplot shapes: geom_point, geom_bar, geom_line, geom_histogram, geom_boxplot, geom_XYZ, geom_density (smooth hist)
Aesthetic attributes: x, y, alpha, color, fill, group, shape, size, stroke. **Ex**. data %>% ggplot(aes(___ = ___)) + geom_()

Reproducibility and Data Viz

Reproducible: if another person can take the same source materials and recreate the same conclusion (needs data, code, docs)
Replicable: if another person create the study in full, collecting new data, and come to the same conclusion.
Associations: Direction (positive/negative), Strength (weak, moderate, strong), Shape (linear, non-linear, exponential)
Facet_wrap(vars(<VAR>)): mult versions of same plot using diff subsets determined by variable. **Bootstrap**: facet_wrap(vars(replicate))
Mapping: dynamic link between the values in a column of your data frame and an aesthetic attribute of your plot (**inside** aes).
Setting: static way to tweak the look of your plot that does not reference the data frame (**outside** aes, in geom)
Labels: labs(title = "title", x = "x", y = "y", color = "what col is color sorted by", Caption = "sources")
Scale: xlim(lower, upper), ylim (outside of all functions)
Overplotting: multiple observations overlap each other, **fix** with jittering, transparency, or diff geometry like hex/contour plot
Jitter: random noise added to both coordinates to separate them from one another + geom_jitter()
Transparency: make points transparent by setting alpha (1 is opaque) + geom_point/geom_jitter(alpha = .1, size = 8)
Themes: + theme_[solarized, wsj, fivethirtyeight, economist, excel]()
Annotation: annotate a plot with lines, text, points, etc. + annotate(geom = "text", label = "Civil war begins", x = 1648, y = 15500)

Subsets

*Always break the pipe after select
Grouping: Using group_by in a pipeline does all calculations after by the group.
group_by() + summarize(): results in a data frame with one group and its stats in each row
Multiple groupings: breaks down by groups listed chronologically

```
class_survey %>%
  group_by(time_at_cal) %>%
  summarize(avg_coding = mean(coding_exp_scale,
                              na.rm = T))
```

```
## # A tibble: 6 × 2
##   time_at_cal                avg_coding
##   <chr>                           <dbl>
## 1 I'm in my first year.            3.21
## 2 I'm in my fourth year.           3.85
## 3 I'm in my second year.           3.02
```

Generalizations/Random Variables

Wrong: based on small samples of unrepresentative data, due to either chance (sampling variability) or systematic bias.
Population: full group of observational units upon which you wish to make a claim (N)
Sample: set you have observed (n, where n <= N), **Census**: n == N, **Anecdote**: n = 1
Probability Distributions: must be disjoint (mutually exclusive), between 0 and 1, must sum to 1.
Parameters: numerical characteristic of a probability distribution or population, often unknown but what we seek to estimate.
Random Variables: random process with numerical outcome, mapping from the outcome space to numbers.
Bernoulli Distribution: two outcomes and p chance of success, Ex: coin flip, water vs land, 0 and 1,
Uniform Distribution: integers between a and b where every outcome is equally likely, **Ex**: dice roll, coin flip
Binomial Distribution: n independent Bernoulli trials, E(X) = np, Var = np(1-p), **Ex**: sum of coin flips
Binomial Coefficient: num ways the binomial outcome can occur
Expected Value: same as mean, E[X] or μ, the sum of all possible outcomes weighted by their probability
E(aX) = aE(x), E(X+Y) = E(X) + E(Y), E(aX+bY) = aE(X) +bE(Y)
Variance: the sum of all squared deviations from expected value weighted by probability, σ^2
Var(aX) = a^2Var(X), Var(X+Y) = Var(X) + Var(Y), Var(aX+bY) = a^2Var(X) + b^2Var(Y)

$$P(Y = y) = \binom{n}{y} p^y (1-p)^{n-y}$$

$$\binom{n}{x} = \frac{n!}{x!(n-x)!}$$

$$E(X) = \sum_{i=1}^{k} x_i P(X = x_i)$$

$$Var(X) = \sum_{i=1}^{k} (x_i - \mu)^2 P(X = x_i)$$

Confidence Intervals/Sampling

An interval constructed from a sample of data that contains the true parameter with 1−α confidence.
Sampling: Replacement has stable populations, as population sizes grow large relative to sample size p remains stable (more w/repl tho)
Ex: If we have 2.5% quantile = 3100, 97.5% quantile = 3232, 95% confidence interval = (LB, UB) = (3100/4947,3232/4947)=(.626,.653)
Bootstrap: method to assess uncertainty, draw a sample of the data that is representative *with replacement*, calculate, repeat many times.
Confidence Intervals: summary of sampling variability in an estimate, percentile bootstrap interval is in the middle 1-α of the bootstrap distribution.
Mathematical methods: math tailored toward certain stats, strong assumptions, requires large n
Binomial Interval: accurate at small n, width of interval (SE) decreases at rate of 1/sqrt(n)
Var(p hat) = 1/n*p(1-p), Var(p hat) = (1/nY)
Percentile Bootstrap: requires reasonable sample size, works for ANY statistic, not just proportions
Simple Random Sampling (SRS): select randomly, each person is indep. and has equally likely chance
Convenience Sampling: people that are easily accessible are more likely to be included in the sample.
Non-response bias: when sampled people don't provide data in a manner that is unrepresentative of the population.

```
boot <- pew %>%
  specify(response = closed,
          success = "yes") %>%
  generate(reps = 500,
           type = "bootstrap") %>%
  calculate(stat = "prop")
ci <- boot %>%
  get_ci(level = .95)
boot %>%
  visualize() +
  shade_ci(ci)
```

Hypothesis Testing

Null Hypothesis: Nothing is going on, variables are independent. **Alt Hypothesis**: Something is going on, they are not indep!
Rejecting Null: if p-value < alpha (usually .05) reject the null, else "fail to reject"! NEVER accept the null hypothesis.
P-value: probability/proportion of test statistics simulated assuming null is true that are more extreme than obs_stat
Steps: state null/alt hyp, calculate observed test statistic, find null dist. Of test stat assuming null is true, find p value and compare to alpha

Permutation

Shuffle or permute the data in one of the variables to generate the kind of data we expect under null hypothesis of independence.
Two-sided p-value: p of data in both directions, get_p_value(obs_stat=obs_stat, direction="both")
Taking Draws: generates data under specific null hyp by taking draws from that dist.
Finding P(outer area): pbinom(q = 25, size = 75, prob = .5)
3 Ways: Simulation/Taking Draws: (taking n draws with success p, converges to exact as reps) increase, can get computationally expensive
Exact method/Binomial: Exact p value using binomial, can get computationally expensive
Approx/CLT: converges to exact p value as n grows large, computationally cheap
**Increasing # of reps stabilizes the distribution and increases precision of p value

```
null <- millennials %>%
  specify(response = response,
          success = "favor") %>%
  hypothesize(null = "point",
              p = .5) %>%
  generate(reps = 500,
           type = "draw") %>%
  calculate(stat = "prop")
null %>%
  visualize() +
  shade_p_value(obs_stat = obs_stat,
                direction = "both")
```

```
null <- promote %>%
  specify(response = decision,
          explanatory = gender,
          success = "promote") %>%
  hypothesize(null = "independence") %>%
  generate(reps = 500,
           type = "permute") %>%
  calculate(stat = "diff in props")
null %>%
  visualize() +
  shade_pvalue(obs_stat = obs_stat,
               direction = "both")
```

```
obs_stat <- yawn %>%
  specify(response = response,
          explanatory = group,
          success = "yawn") %>%
  calculate(stat = "diff in props")
obs_stat
```

Normal distribution: Cont. RV, bell-shaped, centered at μ and with a spread of σ.
*68% of the dist is within 1 SD, 95% within 2 SD, 99.7% within 3 SD.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$$E(X) = \mu$$
$$Var(X) = \sigma^2$$

CLT: sum of indep RVs become normally distributed as n → inf, binomial counts are sums, sample means are norm sums, sample props are means of 0/1s, as n gets larger the distribution becomes smoother, approx by X ~ N(μ=np, capSIG = sqrt(np(1-p)))

Statistical Errors

Hyp tests: used to assess degree to which data is consistent with a particular model + the most widely used tool in statistical inference.
Alt Hyp: default to two-sided tests, **Hypotheses** are statements about true state of the world, involving parameters not stats
4 Ways to Construct Null Dist: Permutation (when null = "independence"), Simulation (when null = "point"), Exact probability theory (when ur lucky), Normal approx (when CLT applies)
Decision Errors: Type 1 (rejected null but null was true), Type 2 (Failed to reject, but null was false)
Affect Error Rates: sample size (inc n dec var), dec alpha (dec type 1 increase type 2), inc effect size (separate HA dist and dec type 2)
Statistical Power: P(reject H_0 | H_0 is false), the probability that you will reject the null hyp if it is false
*the more power, the higher the probability of finding an effect
Observational Study: observes individuals and measures variables of interest but does not attempt to influence the responses.
Experiment: researcher deliberately imposes some treatment on individuals to measure their responses.

```
hyp_0 <- millennials %>%
  specify(response = response, success = "favor") %>%
  hypothesize(null = "point",
              p = .5) %>%
  generate(reps = 9,
           type = "draw") %>%
  ggplot(aes(x = response)) +
    geom_bar(fill = "steelblue") +
    facet_wrap(vars(replicate),
               nrow = 3)
```

```
null_dist <- millennials %>%
  specify(response = response,
          success = "favor") %>%
  hypothesize(null = "point",
              p = .5) %>%
  generate(reps = 500,
           type = "draw") %>%
  calculate(stat = "prop")
null_dist
```

```
null_dist <- yawn %>%
  specify(response = response,
          explanatory = group,
          success = "yawn") %>%
  hypothesize(null = "independence") %>%
  generate(reps = 500,
           type = "permute") %>%
  calculate(stat = "diff in props",
            order = c("no stimulus", "stimulus"))
null_dist
```