GR5065 Assignment 2

Due by 4PM on February 20, 2020

1 Darts

Read this article by Ryan (not Rob) Tibshirani

https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1740-9713.2011.00483.x

on the optimal strategy for playing darts. Then install (once) the Tibshirani's darts package from CRAN outside your .Rmd file, which can produce the dartboard in Figure 1.

```r
library(darts)
drawBoard(new = TRUE)
```

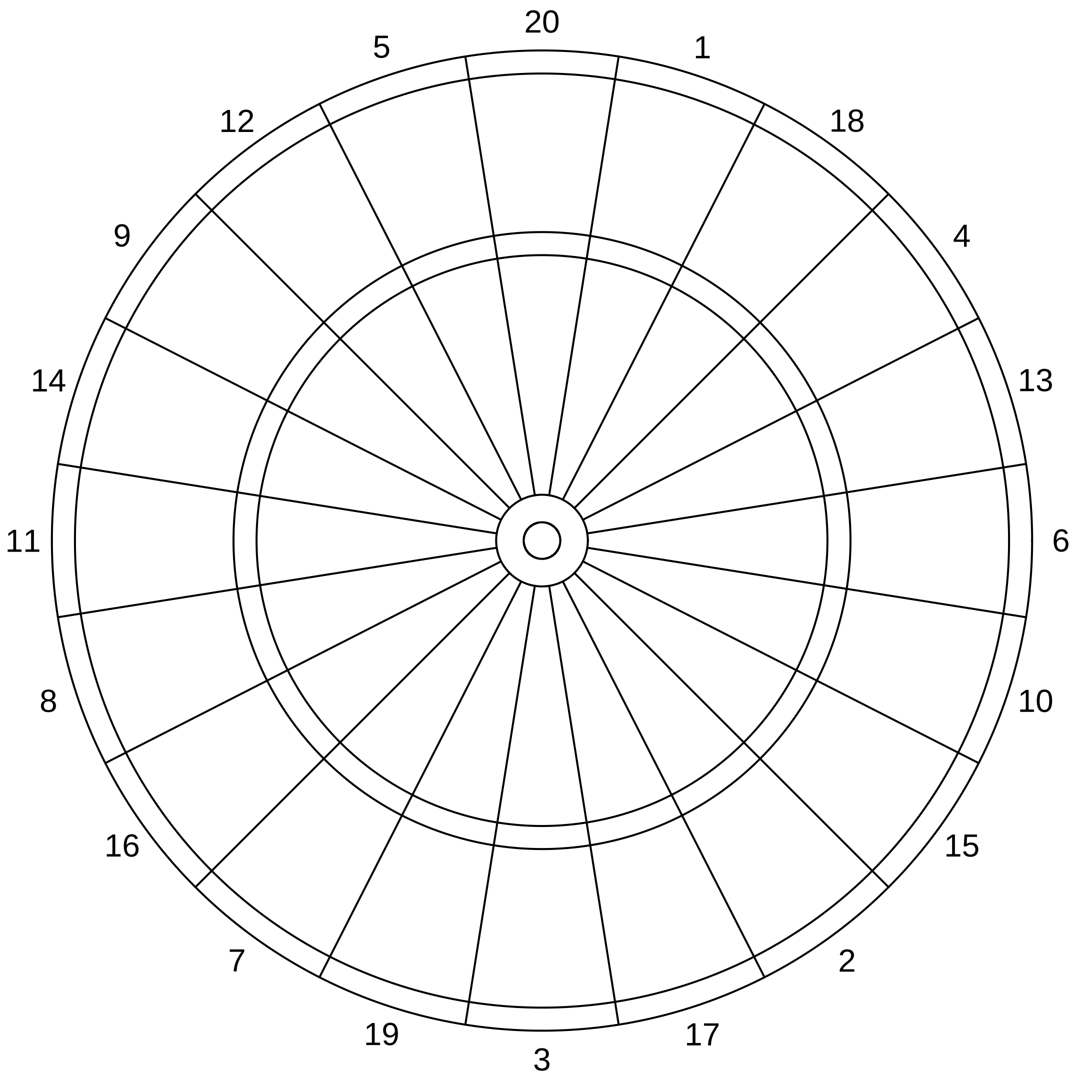

Figure 1: A Conventional Dartboard

If you would like to watch a short YouTube video about throwing darts, one can be found at

https://youtu.be/9gB5ybp_7s8

although it is not at all necessary to answer these problems.

Players throw darts at a circular dartboard, whose measurements are as follows:

- Radius of innermost circle for the "double bullseye": 6.35 mm. If the dart lands in this region, it is worth 50 points.
- Radius of next circle for the "single bullseye": 15.9 mm. If the dart lands in this region, it is worth 25 points.
- Radius of the next circle that starts the triple-score region: 99mm.
- Radius of the next circle that ends the triple-score region: 107mm If the dart lands in this region, it is worth triple the number of points indicated on the outside of the dartboard.
- Radius of the next circle that starts the double-score region: 162mm If the dart lands in this region, it is worth double the number of points indicated on the outside of the dartboard.
- Radius of outermost circle: 170 mm If the dart lands beyond this line, it is worth zero points.

If the dart lands in any other region, it is worth the number of points indicated on the outside of the dartboard.

Thus, the function that maps where the dart lands to the player's score for that throw is both discrete and non-monotonic in the distance from the origin. A function to score the result of a dart throw is as follows, which takes a radius and an angle:

```r
library(dplyr) # for between() function
score <- function(radius, angle) {
  stopifnot(is.numeric(radius), length(radius) == 1, radius >= 0)
  stopifnot(is.numeric(angle),  length(angle) == 1,
            angle  >= -2 * pi, angle < 2 * pi)

  if (radius > 170)   return(0)  # misses dartboard
  if (radius <= 6.35) return(50) # double bullseye
  if (radius <= 15.9) return(25) # single bullseye

  margin <- pi / 20
  interval <- margin * 2

  S <- darts:::getConstants()$S  # 20, 1, ..., 5
  small <- pi / 2 - margin - 0:19 * interval
  large <- pi / 2 + margin - 0:19 * interval
  bed <- which(angle > small & angle <= large)
  if (length(bed) == 0) {
    angle <- angle - 2 * pi
    bed <- which(angle > small & angle <= large)
  }
  score <- S[bed]
  if (between(radius, 99, 107))       score <- 3 * score # in triple ring
  else if (between(radius, 162, 170)) score <- 2 * score # in double ring
  return(score)
}
```

In order to map from Cartesian (X, Y) coordinates to the polar coordinates necessary to call this function, we use

- $r = \sqrt{x^2 + y^2}$
- $\theta = \text{atan2}\,(y, x)$

The inverse mapping from polar coordinates to Cartesian coordinates is

- $x = r\cos\theta$
- $y = r\sin\theta$

Tibshirani's simplest model assumes that the x-coordinate and the y-coordinate for where the dart lands are both distributed normal with expectation zero and unknown standard deviation σ. In other words, the player is aiming for $(0, 0)$. Conditional on σ, X and Y are independent of each other.

1.1 Prior Predictive Distribution

Choose a prior median for what you believe your σ would be if you were to play darts while aiming for $(0, 0)$, keeping in mind that σ is in millimeters. It can also be useful to remember that about 68% of realizations from a normal distribution fall within σ of the expectation.

Then choose a prior lower quartile and a prior upper quartile for your σ. These three pieces of information are sufficient to characterize your prior beliefs about σ under a Johnson Quantile Parameterized Distribution that is Semi-Bounded (JQPDS) at zero (because σ cannot be negative). To access the inverse CDF for the JQPDS, execute

```
library(rstan)
expose_stan_functions("quantile_functions.stan")
args(JQPDS_icdf)
```

```
## function (p, lower_bound, alpha, quantiles, pstream__ = <pointer: 0x7f152c961340>)
## NULL
```

Use the `JQPDS_icdf` functiont to draw *once* from this prior distribution to obtain a realization $\widetilde{\sigma}$. Then, use that $\widetilde{\sigma}$ to draw the x and y-coordinates of $N = 50$ independent dart throws.

Call `darts::drawBoard(new = TRUE)` followed by a call to the `points` function to add all the $\widetilde{x}$ and $\widetilde{y}$ to the plot, like in Figure 4 of Tibshirani's paper. These represent a plausible collection of $N = 50$ dart throws by you.

1.2 Posterior Distribution

Now suppose those x and y-coordinates of $N = 50$ dart throws were actually observed. Use the `curve` function to plot the posterior PDF of $F(\sigma)$ implied by these $N = 50$ data points and your prior. You will need to use the `integrate` function to obtain the denominator of Bayes' Rule. The function to evaluate the numerator of Bayes Rule might look something like

```
numerator <- function(p_vec, x, y, quantiles) {
  return(sapply(p_vec, FUN = function(p) {
    # map p to sigma
    # return the likelihood of sigma (not the log-likelihood)
  }))
}
```

1.3 Sampling Distribution of $\widehat{\sigma}$

Suppose $\sigma = 46$ for some darts player. Suppose further that you plan to observe $N = 50$ dart throws by that player, but only the *score* of each throw will be recorded (using the `score` function above) rather than the x and y-coordinates. Finally, suppose that you plan to use the `darts::simpleEM` function to estimate σ from the scores using the algorithm Tibshirani alludes to in the article (Note that the `$s.final` element of the list returned by `darts::simpleEM` is an estimate of the *variance* so you would need to take the square root of that to estimate the *standard deviation*.)

A function to calculate the sampling distribution of $\widehat{\sigma}$ over essentially all (represented by `S`) ways that N scored dart throws could be collected for this dart player is suggested by

```
sampling_distribution <- function(S, N, sigma) {
  sigma_hat <- rep(NA, S)
  for (s in 1:S) {
    # draw N values of x and y
    # convert each of them to a radius and an angle
    # call the score() function on each radius and angle
    # call the simpleEM() function on the N scores
    # put the square root of the $s.final into sigma_hat[s]
  }
  return(sort(sigma_hat))
}
```

Modify this function so that it actually works and then call it with `S = 10000`, `N = 50`, and `sigma = 46`.

How would you describe the characteristics of Tibshirani's estimator across randomly-sampled datasets on the basis of these `S` simulations?

1.4 Hypothesis Test

The Examples section of the `help(simpleEM, package = "darts")` page contains the following assertion

```
# Scores of 100 of my dart throws, aimed at the center of the board
```

On the basis of the previous subproblem and the above article, would you conclude that these 100 dart throws come from Tibshirani or Price on the basis of a "simple" likelihood-ratio test?

https://en.wikipedia.org/wiki/Likelihood-ratio_test#Case_of_simple_hypotheses

You can ignore `q` because the probability of obtaining a likelhood ratio of exactly `c` is tiny. But you do need to come up with a reasonable value of `c`.

The log-likelihood can be obtained under the hypothesis that the throws were made by Tibshirani or the hypothesis that the throws were made by Price by calling the `simpleEM` function with an appropriate value of the `s.init` argument, which is the initial value of the variance that the algorithm uses. Then, the first element of `$loglik` in the output will contain the corresponding log-likelihood.

1.5 Posterior Distribution

Suppose those 100 dart throws were actually thrown by you. Use prior predictive matching to update your beliefs about σ from the prior that you specified in the first subproblem. Since the probability of observing all 100 of these dart scores will be extremely small, you will need to update your beliefs about σ one dart throw at a time and call the `quantile` function with `probs = c(0.25, 0.5, 0.75)` on the intermediate posterior distributions to form a JQPDS prior that is updated by the score of the next dart throw.

How would you describe the characteristics of the final posterior distribution?

1.6 Optimization

On the basis of the results in the previous subproblem, wow would you decide what is your optimal aiming point on a dartboard if your goal is to maximize your score? You do not actually have to compute it. How does this compare to the mechanism decscribed in the Tibshirani article to compute the optimal aiming point on the basis of $\widehat{\sigma}$?

2 Manipulations of Continuous Probability Distributions

This is a rare case where all of the answers can be given in terms of elementary functions. You can either type out the math (LaTeX is not necessary) or write the math out with a pen and include a picture for each subproblem that you take with your cell phone.

2.1 Parameter Space

Suppose U is distributed standard uniform and, for a given $\kappa > 0$ and $\omega > 0$,

$$\theta = \frac{\kappa}{(1-U)^{\frac{1}{\omega}}}$$

is defined by this inverse Cumulative Distribution Function. What is the parameter space for θ?

2.2 Median

What is an expression for the median of θ?

2.3 Cumulative Density Function

What is an expression for the Cumulative Density Function (CDF) of θ?

2.4 Probability Density Function

What is an expression for the Probability Density Function (PDF) of θ?

2.5 Expectation

What is an expression for the expectation of θ and what additional condition must be satisfied for the expectation to be finite?

2.6 Prior Predictive Density Function

Suppose X is distributed uniform between 0 and θ. What is an expression for the marginal PDF of X?

If it helps, you can think of X as the price in dollars of one Bitcoin at some point in the distant future. It is perhaps reasonable to believe that future price is uniformally distributed between zero and the upper bound for the price of a Bitcoin. But we do not know for sure what the upper bound is.

2.7 Posterior Density Function

Suppose you have one observed $x > 0$. What is an expression for the posterior distribution of θ given x under the previous assumptions?

2.8 Posterior Density Function Given N Observations

Suppose you have N observed values $x_1, x_2, \ldots, x_N$, all of which are positive. What is an expression for the posterior distribution of θ given these N observed values under the previous assumptions?

2.9 Posterior Predictive Density Function

What is the PDF for a *future* $\widetilde{x} > 0$ given the N observed values in the past under the previous assumptions?