1. Functions.

There are two functions below which are missing some or all of the body. The first one should generate data from a linear model. The second should estimate a linear model using an input dataframe and then make some plots to examine the fit.

Complete both functions.

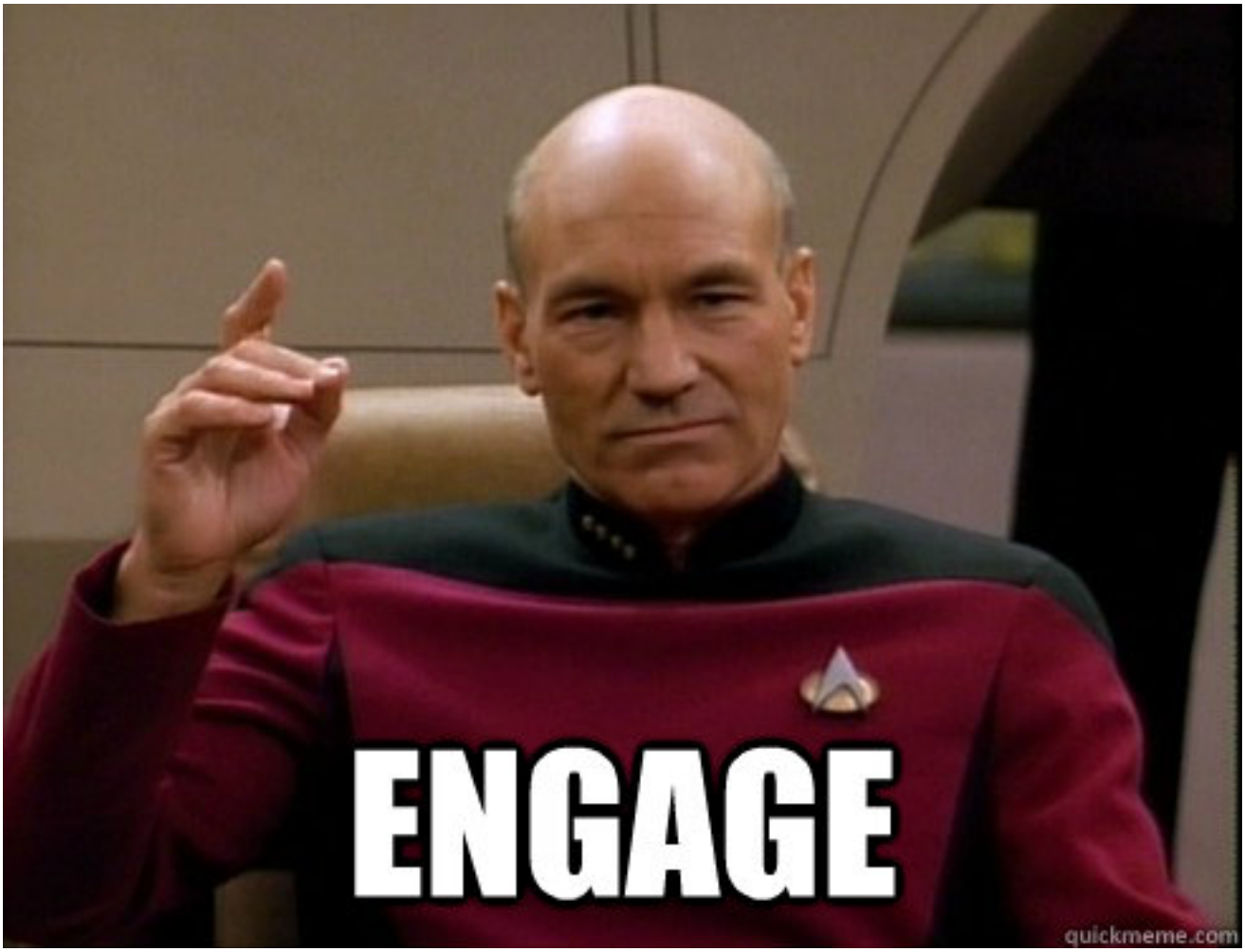
```
generate.data <- function( ,sig.epsilon=1){
  ## you need some more inputs
  ## sig.epsilon - (optional), what is this?
  X = matrix(rnorm(p*n), ncol=p)
  epsilon = rnorm(n, sd = sig.epsilon)
  beta = p:1
  beta.0 = 3
  y =
  df = data.frame(y, X)
  return(df)
}

estimate.and.plot <- function(form, dataframe, plotme = TRUE){
  ## Estimates and (optionally plots some diagnostics for) a linear model
  ## Takes in a formula, as formula('y~x') or somesuch
  ## and data frame
  ## plotme determines ...
  mdl = lm(form, data=dataframe)
  if(plotme){
    preds = labels(terms(form, data=dataframe))
    df = dataframe[preds]
    df$resids =  # how do you get residuals?
    df$fit =  # how do you get the fitted values?
    preds.vs.resids = df %>%
      gather(-c(resids,fit), key='predictor', value='value')
    # create a new dataframe for ggplot
    # what does this do?
    p1 <- ggplot(preds.vs.resids, aes(x=value, y=resids)) + geom_point() +
      geom_smooth() + facet_wrap(~predictor, scales = 'free')
      # ??
    p2 <- ggplot(df, aes(sample=resids)) + geom_qq() + geom_qq_line()
    # ??
    print(p1) # print out the first plot (wouldn't do this inside a function generally)
    print(p2) # print out the second plot
  }
  return(mdl) # output our fitted model
}
```

2. Function execution.

- Generate some data with the first function. Use 4 predictors (you can choose n and the noise SD yourself).

- Estimate the model with the second function. And produce the plots.
- Create a table which shows the coefficients, their standard errors, and p-values. You must use the `knitr::kable` function to do this. Print only 2 significant digits. Hint: there is a way to extract all of this information easily from the `lm` output.

3. Engage.



You will now attempt to re-engage last semester's brain cells by doing things you should already know how to do in possibly new ways. Consider the "properties.txt" dataset from HW 9 and 10 (optional) in S431. Recall that it has an outcome (rental rates) and four predictors (age, operating expenses + taxes, vacancy rates, square footage). The goal is to predict rental rates using these four variables.

1. Use the `lm` function to estimate the linear model of rental rates on all four predictors. Produce a table summarizing the output.
2. Make plots of the residuals against each predictor. Make a qq-plot of the residuals. Discuss what you see. Does the assumption of "normally distributed residuals" appear to be satisfied?
3. Interpret the estimated coefficient on vacancy rates. Find and interpret a 90% confidence interval for $\beta_{vacancy}$. Test, with $\alpha = 0.05$, whether or not $\beta_{vacancy} = 0$. State your conclusion in the context of the problem.

4. Someone suggests including an interaction for age and vacancy rates. Add this interaction to the model reinterpret the effect of vacancy rates on rental rates.

5. Someone suggests that it would be better to use the log of rental rates as the outcome. Repeat steps 1 to 3 with this change.