# Homework 2

Advanced Methods for Data Analysis (36-402)

Due Friday February 4, 2022 *at 3:00 pm*

**Solutions** – not to be posted online or shared, even after the end of the semester.

You should **always show all your work** and submit both a writeup and *R* code.

- Assignments must be submitted through Gradescope as a PDF. Follow the instructions here: https://www.cmu.edu/teaching/gradescope/
- Gradescope will ask you to mark which parts of your submission correspond to each homework problem. This is mandatory; if you do not, grading will be slowed down, and your assignment will be penalized.
- Make sure your work is legible in Gradescope. You may not receive credit for work the TAs cannot read. **Note:** If you submit a PDF with pages much larger than $8.5 \times 11$", they will be blurry and unreadable in Gradescope.
- For questions involving R code, we strongly recommend using R Markdown. The relevant code should be included with each question, rather than in an appendix. A template Rmd file is provided on Canvas.

1. **A Refresher in Linear Regression.** The data for this problem are in two files named `housetrain.csv` (containing training data) and `housetest.csv` (containing test data.) They are both comma-separated files with headers. You will need to read each of them into an *R* `data.frame`. The `read.csv` command will do this, if you get the syntax correct.

   The data are from a census survey from several years ago. Each record (line in the file) corresponds to a small area called a *census tract*. The variables that appear in the data file have the following names:

   - `Population`: The population of the census tract.
   - `Latitude`: The number of degrees north of the equator where the census tract is located. South is negative, so latitude is between $-90$ and 90.
   - `Longitude`: The number of degrees east of Greenwich where the census tract is located. West is negative, so longitude is between $-180$ and 180.
   - `Median_house_value`: The median assessed value of houses (in thousands of dollars) in the census tract.
   - `Median_household_income`: The median household income in the census tract.
   - `Mean_household_income`: The average household income in the census tract.

   The data are from census tracts in California and Pennsylvania.

   The main goal of this problem is to model the relationship, if any, between `Median_house_value` (the response) and the other variables (potential predictors.) For all plots, use the option `pch="."` because the default circles will overlap too much with such large data sets.

(a) Compute the correlation matrix between all of the variables in the training data set. Which potential predictors are most highly correlated (positively or negatively) with the response?

**Solution:**

```
set.seed(1000)
dat <- read.csv("housetrain.csv", header = TRUE)
cor(dat)
```

```
##                           Population   Latitude  Longitude Median_house_value
## Population                 1.00000000 -0.1614544 -0.1613919         0.05371813
## Latitude                  -0.16145444  1.0000000  0.7281800        -0.46582911
## Longitude                 -0.16139190  0.7281800  1.0000000        -0.53908709
## Median_house_value         0.05371813 -0.4658291 -0.5390871         1.00000000
## Median_household_income    0.12184024 -0.1354334 -0.1835474         0.64652324
## Mean_household_income      0.08612980 -0.1248609 -0.1714566         0.69413825
##                           Median_household_income Mean_household_income
## Population                              0.1218402             0.0861298
## Latitude                               -0.1354334            -0.1248609
## Longitude                              -0.1835474            -0.1714566
## Median_house_value                      0.6465232             0.6941383
## Median_household_income                 1.0000000             0.9492757
## Mean_household_income                   0.9492757             1.0000000
```

Not surprisingly, Mean_household_income and Median_household_income are strongly positively correlated with the response.

(b) Use the training data to fit the following models:

Model 0: The *null model*, which says that conditional on all of the potential predictors, the values of $Y_j$ are independent and identically distributed with some mean $\mu$.

Model 1: A simple linear regression of the response on `Median_household_income`.

Model 2: A simple linear regression of the response on `Mean_household_income`.

Model 3: A multiple regression of the response on *both* `Median_household_income` and `Mean_household_income`.

Model 4: A regression of the response on `Median_household_income`, `Mean_household_income` and 5 simulated covariates of your choice that are *independent* of the response. (In other words, create five new columns in the data frame and fill them with random variables independent of the response variable.) Briefly mention how you simulated these extra covariates.

For each model, include a summary and examine its residuals. Based on the residuals and your diagnostics, do you think the underlying assumptions are reasonable? Give a brief justification.

**Solution:** In Model 4, to create simulated covariates that are independent of the response, we simply create five columns of normal random variables and add them into the data frame. Any random data will do; it doesn't matter much if it's normal, uniform, or anything else, as long as it's independent of the response.

```
fit0 <- lm(Median_house_value ~ 1, data = dat)
fit1 <- lm(Median_house_value ~ Median_household_income, data = dat)
fit2 <- lm(Median_house_value ~ Mean_household_income, data = dat)
fit3 <- lm(Median_house_value ~ Median_household_income + Mean_household_income,
           data = dat)
sim_dat <- cbind(dat, matrix(rnorm(nrow(dat) * 5), ncol = 5, nrow = nrow(dat)))
colnames(sim_dat) <- c(colnames(dat), paste0("X", 1:5))
fit4 <- lm(Median_house_value ~ Median_household_income + Mean_household_income +
             X1 + X2 + X3 + X4 + X5, data = sim_dat)
```

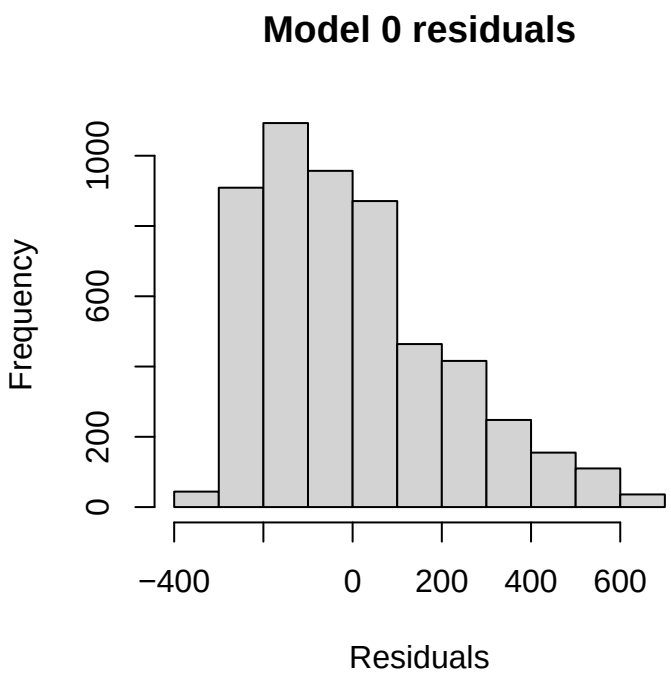
Next, we present the summary of the models and some diagnostics, namely 1) scatter plots of the residuals versus each predictor, 2) a scatter plot of residuals versus fitted values, 3) scatter plot of squared residuals versus fitted values, 4) distribution of the residuals.

For Model 0, we just present the distribution of the residuals.

```
summary(fit0)
```

```
##
## Call:
## lm(formula = Median_house_value ~ 1, data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -328.26 -170.26  -36.56  114.69  653.54
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  344.455      2.891   119.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 210.5 on 5302 degrees of freedom
```

```
hist(residuals(fit0),
     xlab="Residuals", main="Model 0 residuals")
```



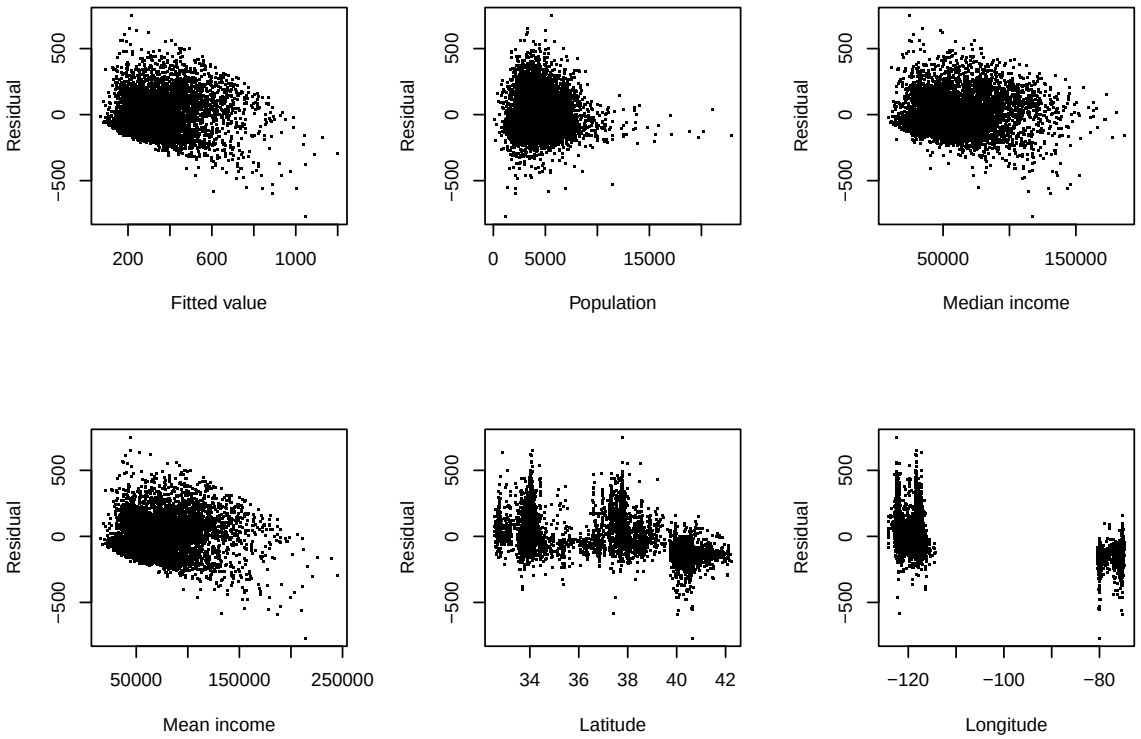

Notice that the distribution of the residuals for model 0 presents a long right tail. This is to be expected because we are modeling $Y$ with its mean. So the residuals are just a centered version of $Y$. Generally, when the variable represents money, it is rarely normally distributed, rather it presents long tails. We need to move towards richer models.

We then present results for Model 3. Residuals plots for models 1, 2, and 4 are similar, so we do not show them here.

```
summary(fit3)
```

```
##
## Call:
## lm(formula = Median_house_value ~ Median_household_income + Mean_household_income,
##     data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -770.87 -112.34  -21.68   97.43  749.98
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -3.4325926  5.3880615  -0.637    0.524
## Median_household_income -0.0010032  0.0002511  -3.996 6.53e-05 ***
## Mean_household_income    0.0054576  0.0002107  25.901  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 151.3 on 5300 degrees of freedom
## Multiple R-squared:  0.4834,Adjusted R-squared:  0.4832
## F-statistic:  2480 on 2 and 5300 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2, 3))
plot(fitted(fit3), residuals(fit3), pch=".",
     xlab="Fitted value", ylab="Residual")

plot(dat$Population, residuals(fit3), pch=".",
     xlab="Population", ylab="Residual")

plot(dat$Median_household_income, residuals(fit3), pch=".",
     xlab="Median income", ylab="Residual")

plot(dat$Mean_household_income, residuals(fit3), pch=".",
     xlab="Mean income", ylab="Residual")

plot(dat$Latitude, residuals(fit3), pch=".",
     xlab="Latitude", ylab="Residual")

plot(dat$Longitude, residuals(fit3), pch=".",
     xlab="Longitude", ylab="Residual")
```




Residuals plots for model3 show a relationship between residuals and fitted in terms of mean (resid v. fitted and v. each predictor) and variance. This suggests violations of the assumptions that $\mathbb{E}[\epsilon|X] = 0$ and $\text{Var}(Y|X)$ is constant. Also, the residuals do not look normally distributed (although this violation does not appear too severe), but this is less important, given that we first need to take of the violations of non-zero mean and non-constant variance.

Also, notice how the residuals vary with longitude and latitude (after all, we are dealing with PA and CA). This suggests including these variables into the model.

(c) Explain why the coefficients of `Median_household_income` and `Mean_household_income` in Model 3 are both different from the coefficients of the same predictors in Models 1 and 2.

**Solution:** The correlation between the two income variables is quite high, so the presence of each one can be expected to affect the estimated coefficient for the other one.

(d) How does the *training error* for Model 4 compare to the *training error* in Model 3? What about the $R^2$? What do you expect that would happen to the training error and $R^2$ if you include more and more covariates that are independent of the response?

**Solution:** First we compare training error:

```
resids_train <- c(mean(resid(fit0)^2), mean(resid(fit1)^2),
                  mean(resid(fit2)^2), mean(resid(fit3)^2), mean(resid(fit4)^2))
names(resids_train) <- paste0("resid_train", 0:4)
sort(resids_train)
```

```
## resid_train4 resid_train3 resid_train2 resid_train1 resid_train0
##     22861.73     22891.66     22960.63     25789.24     44310.82
```

Next, the $R^2$ (not adjusted):

```
summary(fit3)$r.squared
```

```
## [1] 0.4833843
```

```
summary(fit4)$r.squared
```

```
## [1] 0.4840599
```

The fit in Model 4 appears to be better than that of Model 3, but this is weird since the additional 5 variables are not related to the response in any way. This is not surprising though, in finite sample, one gets very mild associations between variables just because of noise (variables are never exactly orthogonal in finite samples). When we fit a model via optimization, we essentially pick up some noise too, and that's why it seems we keep improving the error by adding more and more variables. This is why you should never consider the training error as a good measure of model performance, rather one should use a test set.

(e) Compute the "test error" for each model (average squared error from predicting the test responses using the test predictors and the model fit with training data). Make a table showing the training and testing errors of all the models. Which of the models are best? Discuss based on the test error and the analysis from Question (b).

**Solution:** To calculate the test error for Model 4, we have to deal with the five covariates we added into the training data. Here we will simulate five entirely new columns of random data and add it to the test dataset. (You should use new random data, not reuse the random numbers you made for the training data: the test data is meant to be *new* data drawn from the same distribution as the old.)

```
testdat <- read.csv("housetest.csv", header = TRUE)
ytest <- testdat$Median_house_value
resid0 <- mean((ytest - predict.lm(fit0, newdata = testdat))^2)
resid1 <- mean((ytest - predict.lm(fit1, newdata = testdat))^2)
resid2 <- mean((ytest - predict.lm(fit2, newdata = testdat))^2)
resid3 <- mean((ytest - predict.lm(fit3, newdata = testdat))^2)

sim_testdat <- cbind(testdat, matrix(rnorm(nrow(testdat) * 5),
                                     ncol = 5, nrow = nrow(testdat)))
colnames(sim_testdat) <- c(colnames(testdat), paste0("X", 1:5))
resid4 <- mean((ytest - predict.lm(fit4, newdata = sim_testdat))^2)
resids <- c(resid0, resid1, resid2, resid3, resid4)
names(resids) <- paste0("resid", 0:4)
sort(resids)
```

```
##   resid3   resid4   resid2   resid1   resid0
## 22813.26 22814.78 22985.77 26359.29 44062.46
```

Model 3 appears to outperform the other models. Notice how model 4, the best in terms of training error, is doing worse than model 3. Also, notice that the drop in test error between model 1 (one variable) and model 3 (2 variables) is not very large. This might suggest we want to penalize a model for having too many variables; we would enter in the world of model selection (e.g. procedures such as cross-validation, AIC, or BIC, among many others).

2. **Relaxing Our Regression Assumptions.** Consider arbitrary random variables $X \in \mathbb{R}^p$, $Y \in \mathbb{R}$ with *absolutely no assumptions relating the two*, and consider linearly regressing $Y$ on $X$ (in the population), with regression coefficients defined by

$$\beta = \mathrm{Var}(X)^{-1}\mathrm{Cov}(X, Y), \quad \beta_0 = \mathbb{E}(Y) - \beta^T \mathbb{E}(X).$$

Conditional on $X$, our prediction for $Y$ is hence $\beta_0 + \beta^T X$.

(a) Define the residual error $\epsilon = Y - \beta_0 - \beta^T X$. Prove that $\epsilon$ has mean zero, $\mathbb{E}(\epsilon) = 0$. Again, you are only allowed to use the definitions of $\beta$ and $\beta_0$ above and properties of expectations.

**Solution:** Notice that $\beta$ and $\beta_0$ are not random quantities, that is the expectation is taken with respect to $(Y, X)$. By substitution:

$$\begin{aligned}
\mathbb{E}[\epsilon] &= \mathbb{E}[Y - \beta_0 - \beta^T X] \\
&= \mathbb{E}[Y] - \mathbb{E}[\beta_0] - \beta^T \mathbb{E}[X] \\
&= \mathbb{E}[Y] - \mathbb{E}[Y] + \beta^T \mathbb{E}[X] - \beta^T \mathbb{E}[X] \\
&= 0.
\end{aligned}$$

Notice then that $\mathbb{E}[\epsilon] = 0$ is not really an assumption, since the intercept is taking care of the centering. However, the stronger condition $\mathbb{E}[\epsilon | X = x] = 0$ for any $x \in \mathrm{supp}(X)$ is an assumption, as it might not hold in practice.

(b) Prove that $\epsilon$ is uncorrelated with the predictor variables, $\mathrm{Cov}(\epsilon, X) = 0$.

*Hint: Make sure you remember how to manipulate matrices and their transposes, and review your properties of expectations and variances. You can use that* $\mathrm{Cov}(Y, X)^T = \mathrm{Cov}(X, Y)$ *without a proof.*

**Solution:**

$$\begin{aligned}\mathrm{Cov}(\epsilon, X) &= \mathrm{Cov}(Y - \beta_0 - \beta^T X, X) \\ &= \mathrm{Cov}(Y, X) - \mathrm{Cov}(\beta_0, X) - \beta^T \mathrm{Var}(X) \\ &= \mathrm{Cov}(Y, X) - 0 - \beta^T \mathrm{Var}(X) \\ &= 0.\end{aligned}$$

We first substituted in the definition, $\epsilon = Y - \beta_0 - \beta^T X$. Then we used the definition that $\beta = \mathrm{Var}(X)^{-1}\mathrm{Cov}(X, Y)$; the variance matrices cancel out, leaving us with the covariance minus itself.

Also note that because $\beta_0$ is a constant, $\mathrm{Cov}(\beta_0, X) = 0$.

(c) By construction, we have the relationship $Y = \beta_0 + \beta^T X + \epsilon$, i.e., we've written $Y$ as a linear function of $X$ plus an error term $\epsilon$. This error term has mean zero by part (a). Does part (b) imply that the error term is independent of $X$? What in particular does this mean about the conditional variance $\mathrm{Var}(\epsilon|X)$? Need this be constant with $X$?

**Solution:** Recall that $\mathrm{Cov}(X, Y) = 0$ implies independence only if $(X, Y)$ are *jointly* normally distributed. Notice that two random variables could be marginally normally distributed but not jointly, in which case the implication does not hold. In case you are interested, check the Wiki at https://en.wikipedia.org/wiki/Normally_distributed_and_uncorrelated_does_not_imply_independent.

So, even if $\mathrm{Cov}(\epsilon, X) = 0$, $\epsilon \not\perp\!\!\!\perp X$ in general. In particular, notice that

$$\begin{aligned}\mathrm{Var}(\epsilon|X) &= \mathrm{Var}(Y - \beta_0 - \beta^T X|X) \\ &= \mathrm{Var}(Y|X)\end{aligned}$$

For example, let $Y \sim N\left(\beta_0 + \beta^T X, \mathbb{1}\{X_1 \geq 0\} + 2\mathbb{1}\{X_1 < 0\}\right)$, that $Y$ is twice as more variable if $X_1$ is negative. Again, $\mathrm{Cov}(\epsilon, X) = 0$, but $\mathrm{Var}(Y|X) = \mathbb{1}\{X_1 \geq 0\} + 2\mathbb{1}\{X_1 < 0\}$ so it varies with $X_1$.

However, if $\epsilon \perp\!\!\!\perp X$, then

$$\begin{aligned}\mathrm{Var}(\epsilon|X) &= \mathbb{E}\left[\left(\epsilon - \mathbb{E}\left\{\epsilon|X\right\}\right)^2 |X\right] \\ &= \mathbb{E}\left[\epsilon\epsilon^T\right] \\ &= \mathrm{Var}(\epsilon),\end{aligned}$$

which is a constant.

(d) Consider i.i.d. data $(X_i, Y_i)$, $i = 1, \ldots n$, each with the same distribution as $(X, Y)$. For simplicity you may assume from now on that $\mathbb{E}(X) = \mathbb{E}(Y) = 0$ (though this is not really necessary). Use the same sample notation as in Lecture, i.e., $\vec{Y} = (Y_1, \ldots Y_n)^T \in \mathbb{R}^n$ for the vector of outcomes, and

$$\mathbb{X} = \begin{pmatrix} X_1^T \\ X_2^T \\ \vdots \\ X_n^T \end{pmatrix} \in \mathbb{R}^{n \times p}$$

for the design matrix for a sample of size $n$.

Consider the least squares estimator

$$\hat{\beta} = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\vec{Y}.$$

Compute $\mathbb{E}(\hat{\beta}|X^n)$, where $X^n = \{X_1, ....X_n\}$ is the set of inputs. Is $\mathbb{E}(\hat{\beta}|X^n)$ necessarily equal to $\beta = \text{Var}(X)^{-1}\text{Cov}(X, Y)$? If not, under what assumptions will it be?

*Hint: Conditional on the inputs $X^n$, you can treat the design matrix $\mathbb{X}$ as a constant matrix; this is the "fixed design" setting.*

**Solution:** Notice that $\mathbb{E}[Y] = \mathbb{E}[X] = 0$ implies that $\beta_0 = 0$.

$$\begin{aligned}\mathbb{E}\left[\hat{\beta} \mid X^n\right] &= \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T\left(\mathbb{X}\beta + \mathbb{E}[\epsilon \mid X^n]\right) \\ &= \beta + \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T\mathbb{E}[\epsilon \mid X^n]\end{aligned}$$

which is equal to $\beta$ if $\mathbb{E}[\epsilon|X^n] = 0$, which would occur for example if $\epsilon \perp\!\!\!\perp X$.

(e) Compute $\text{Var}(\hat{\beta}|X^n)$. In your formula, you can denote the variance of $\vec{\epsilon} = \vec{Y} - \mathbb{X}\beta$ conditional on $X_1, \ldots, X_n$ by $\text{Var}(\vec{\epsilon} \mid X^n) = \Sigma$. What does your formula reduce to in the case that $\Sigma = \sigma^2 I$, where $I$ is the $n \times n$ identity matrix (i.e., the case where the $\epsilon_i$'s have the same variance)?

**Solution:** Notice that $\text{Var}(\epsilon + \mathbb{X}\beta|X^n) = \text{Var}(\epsilon|X^n)$ because of the conditioning on $X^n$ (otherwise it would not be true).

$$\begin{aligned}\text{Var}\left(\hat{\beta} \mid X^n\right) &= \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T\text{Var}(\epsilon \mid X^n)\mathbb{X}\left(\mathbb{X}^T\mathbb{X}\right)^{-1} \\ &= \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T\Sigma\mathbb{X}\left(\mathbb{X}^T\mathbb{X}\right)^{-1}\end{aligned}$$

If $\Sigma = \sigma^2 I$, then we can cancel out terms:

$$\text{Var}\left(\hat{\beta} \mid X^n\right) = \sigma^2\left(\mathbb{X}^T\mathbb{X}\right)^{-1}$$