Problem 1

These are questions about Silver's book, chapters 3-6. For all parts in this question, answer using notation from class (i.e. $t, f, g, h^*, \delta, \epsilon, e, t, z_1, \ldots, z_t, \mathbb{D}, \mathcal{H}, \mathcal{A}, \mathcal{X}, \mathcal{Y}, X, y, n, p, x_{\cdot 1}, \ldots, x_{\cdot p}$, $x_{1\cdot}, \ldots, x_{n\cdot}$, etc.

(a) [difficult] Chapter 4 is all about predicting weather. Broadly speaking, what is the problem with weather predictions? Make sure you use the framework and notation from class. This is not an easy question and we will discuss in class. Do your best.
The world is 3d, and to improve weather forecast you need to solve the behavior of molecules, and to represent the molecules you need to use a grid which is 2d. Also the grids that are used are 200 x 200 and you use those to predict the weather in one grid. But 200 miles is a big gap and there can be vastly different weather on one side of the grid vs the other.

(b) [easy] Why does the weatherman lie about the chance of rain? And where should you go if you want honest forecasts? You should go to national weather services.
The weatherman will lie and say there is a higher percent chance of rain then there actually is bc if he says it's going to rain, but it doesn't, then it's an amazing treat. But if he says it's not going to rain, but it does, then he ruins everyones day.

(c) [difficult] Chapter 5 is all about predicting earthquakes. Broadly speaking, what is the problem with earthquake predictions? It is *not* the same as the problem of predicting weather. Read page 162 a few times. Make sure you use the framework and notation from class. Weather is easier to predict bc meteorologists have a better understanding of the Earth's atmosphere then seismologists have of the Earth's crust bc if seismologists want to study the crust, they have to dig 15 km underground (which is impossible).

(d) [easy] Silver has quite a whimsical explanation of overfitting on page 163 but it is really educational! What is the nonsense predictor in the model he describes?
The combinations to the red, black, and blue locks

(e) [easy] John von Neumann was credited with saying that "with four parameters I can fit an elephant and with five I can make him wiggle his trunk". What did he mean by that and what is the message to you, the budding data scientist? He was saying that if he's given all of these predictors, he can connect the data points in a way that is aesthetically pleasing, but mathematically nonsense.
Message to me is don't compromise the actual usefulness of your model, by overfitting just bc it will look good on paper. You have to take the L sometimes.

(f) [difficult] Chapter 6 is all about predicting unemployment, an index of macroeconomic performance of a country. Broadly speaking, what is the problem with unemployment predictions? It is *not* the same as the problem of predicting weather or earthquakes. Make sure you use the framework and notation from class.

(g) [E.C.] Many times in this chapter Silver says something on the order of "you need to have theories about how things function in order to make good predictions." Do you agree? Discuss.

Problem 2

These are questions related to the concept of orthogonal projection, QR decomposition and its relationship with least squares linear modeling.

(a) [easy] Let $\boldsymbol{H}$ be the orthogonal projection onto colsp $[\boldsymbol{X}]$ where $\boldsymbol{X}$ is a $n \times (p+1)$ matrix with all columns linearly independent from each other. What is rank $[\boldsymbol{H}]$?

$p+1$

(b) [easy] Simplify $\boldsymbol{HX}$ by substituting for $\boldsymbol{H}$.

$H = X(X^TX)^{-1}X^T \rightarrow HX = X(X^TX)^{-1}X^TX \rightarrow (X^TX)^{-1} \cdot X^TX = I \rightarrow$ so $\therefore HX = XI \rightarrow \boxed{HX = X}$

(c) [harder] What does your answer from the previous question mean conceptually?

Since we're projecting something onto something that it's already on, we're gonna get the same thing back. You can't project it anymore

(d) [difficult] Let $\boldsymbol{X}'$ be the matrix of $\boldsymbol{X}$ whose columns are in reverse order meaning that $\boldsymbol{X} = [\mathbf{1}_n \vdots \boldsymbol{x}_{\cdot 1} \vdots \ldots \vdots \boldsymbol{x}_{\cdot p}]$ and $\boldsymbol{X}' = [\boldsymbol{x}_{\cdot p} \vdots \ldots \vdots \boldsymbol{x}_{\cdot 1} \vdots \mathbf{1}_n]$. Show that the projection matrix that projects onto colsp $[X]$ is the same exact projection matrix that projects onto colsp $[X']$. $H = X(X^TX)^{-1}X^T$ and $H' = X'(X'^TX')^{-1}X'^T$, since the content inside each column is the same, just the columns are reversed, the linear dependencies is still there and $\therefore X^TX = X'^TX' \rightarrow (X^TX)^{-1} = (X'^TX')^{-1}$ and now if we substitute these back in, we get $H = X(X^TX)^{-1}X^T$ and $H' = X'(X^TX)^{-1}X'^T$ and now we can say their colspace is the same so H and H' both project onto colsp(X)

(e) [difficult] [MA] Generalize the previous problem by proving that orthogonal projection matrices that project onto any specific subspace are *unique*.

(f) [difficult] [MA] Prove that if a square matrix is both symmetric and idempotent then it must be an orthogonal projection matrix.

(g) [easy] Prove that I_n is an orthogonal projection matrix $\forall n$.

by def, $I_n \times I_n$ is I_n, and I_n^T is also I_n bc it's diagonal so any diagonal matrix^T is that matrix. Since I_n is idempotent and symmetric, it satisfies the condition for orthogonal projection matrix

(h) [easy] What subspace does I_n project onto?

onto the entire nth dimension, $\mathbb{R}^n$.

(i) [easy] Consider least squares linear regression using a design matrix X with rank $p+1$. What are the degrees of freedom in the resulting model? What does this mean?

The degrees of freedom are p+1 bc that's the amount of independent predictor coefficients there are.

(j) [easy] If you are orthogonally projecting the vector $\boldsymbol{y}$ onto the column space of X which is of rank $p+1$, derive the formula for $\text{Proj}_{\text{colsp}[X]}[\boldsymbol{y}]$. Is this the same as in OLS?

$H = X(X^TX)^{-1}X^T \rightarrow \text{proj}_{\text{colsp}(X)}(y) = Hy = X(X^TX)^{-1}X^Ty$

$b = (X^TX)^{-1}X^Ty$ bc this will minimize the squared differences

$\rightarrow \hat{y} = Xb = X(X^TX)^{-1}X^Ty$ which is the same as our H, $\therefore$ they are the same

(k) [difficult] We saw that the perceptron is an *iterative algorithm*. This means that it goes through multiple iterations in order to converge to a closer and closer $\boldsymbol{w}$. Why not do the same with linear least squares regression? Consider the following. Regress $\boldsymbol{y}$ using $\boldsymbol{X}$ to get $\hat{\boldsymbol{y}}$. This generates residuals $\boldsymbol{e}$ (the leftover piece of $\boldsymbol{y}$ that wasn't explained by the regression's fit, $\hat{\boldsymbol{y}}$). Now try again! Regress $\boldsymbol{e}$ using $\boldsymbol{X}$ and then get new residuals $\boldsymbol{e}_{new}$. Would $\boldsymbol{e}_{new}$ be closer to $\boldsymbol{0}_n$ than the first $\boldsymbol{e}$? That is, wouldn't this yield a better model on iteration #2? Yes/no and explain.

(l) [harder] Prove that $\boldsymbol{Q}^\top = \boldsymbol{Q}^{-1}$ where $\boldsymbol{Q}$ is an orthonormal matrix such that colsp $[\boldsymbol{Q}] =$ colsp $[\boldsymbol{X}]$ and $\boldsymbol{Q}$ and $\boldsymbol{X}$ are both matrices $\in \mathbb{R}^{n \times (p+1)}$ and $n = p + 1$ in this case to ensure the inverse is defined. Hint: this is purely a linear algebra exercise and it's a one-liner. $Q^TQ = I_n \quad QQ^{-1} = I_n \longrightarrow QQ^{-1} = QQ^T \longrightarrow Q^{-1} = Q^T$

(m) [easy] Prove that the least squares projection $\boldsymbol{H} = \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T = \boldsymbol{Q}\boldsymbol{Q}^\top$. Justify each step. I showed before that $QQ^T = I_n \longrightarrow X(X^TX)^{-1}X^T = I_n$ and since w/ least squares, we're dealing w/ orthonormal matrices, as well, $X^TX = I_n$

$\longrightarrow X(I_n)^{-1}X^T = I_n \longrightarrow I_n^{-1} = I_n \longrightarrow XI_nX^T = I_n \longrightarrow XI_n = X$

$\longrightarrow XX^T = I_n \longrightarrow XX^T = I_n \longrightarrow \boxed{I_n = I_n}$

(n) [difficult] [MA] This problem is independent of the others. Let H be an orthogonal projection matrix. Prove that rank $[\boldsymbol{H}]$ = tr $[\boldsymbol{H}]$. Hint: you will need to use facts about eigenvalues and the eigendecomposition of projection matrices.

(o) [harder] Prove that an orthogonal projection onto the colsp $[Q]$ is the same as the sum of the projections onto each column of Q.

$H = Q(Q^TQ)^{-1}Q^T$ but since Q is orthonormal, $Q^TQ = I_n$ and $I_n^{-1} = I_n$

$\rightarrow H = QQ^T \rightarrow Hv = QQ^Tv \rightarrow QQ^T = q_1q_1^T + q_2q_2^T + \dots q_nq_n^T$

So Hv can be written as $Hv = (q_1q_1^T + q_2q_2^T + \dots q_nq_n^T)v = q_1q_1^Tv + q_2q_2^Tv + \dots + q_nq_n^Tv$

(p) [easy] Explain why adding a new column to X results in no change in the SST remaining the same. bc SST relies on y and $\bar{y}$ so adding new predictor columns will just make our y and $\bar{y}$ the same

(q) [harder] Prove that adding a new column to X results in SSR increasing.

bc when we add a new column we just get more variability in y and bc OLS minimizes the e's, it will maximize SSR. And w/ this new predictor, it will increase SSR bc there's more variability in y

(r) [harder] What is overfitting? Use what you learned in this problem to frame your answer. It's when you try to hard to capture every detail in the training data. So when you add too many predictor columns, it may look like the model is fitting really well, but it's just focused on the noise

(s) [easy] Why are "in-sample" error metrics (e.g. R^2, SSE, s_e) dishonest? Note: I'm leaving out RMSE as RMSE attempts to be honest by increasing as p increases due to the denominator. I've chosen to use standard error of the residuals as the error metric of choice going forward. Bc the more predictors you add, it will fit everything including the noise very well and it will be overly optimistic. Adding new predictors, especially ones that don't add anything meaningful can also make the model unnecessarily complex

(t) [easy] How can we provide honest error metrics (e.g. R^2, SSE, s_e)? It may help to draw a picture of the procedure. You can split your data into testing and training and then you can train your data using the training data but calculate it using the testing data. Or you can use cross validation by dividing your D into k folds, and then for each round you can train the model on k-1 folds, and test it on the remaining fold

(u) [easy] The procedure in (t) produces highly variable honest error metrics. Can you change the procedure slightly to reduce the variation in the honest error metrics? What is this procedure called and how is it done? K fold cross validation. Split your ID into K folds, use the K-1 folds as a training set, and the last fold as a test set. Find the error metrics for each iteration and average the error metrics

Problem 3

These are some questions related to validation.

(a) [easy] Assume you are doing one train-test split where you build the model on the training set and validate on the test set. What does the constant K control? And what is its tradeoff? K is the percentage of your ID that is used for the test set. If you have a larger K, it will give you a better estimate on how your model will perform on unseen data, but it will leave less data for training which can lead to a less trained model. And vice versa

(b) [harder] Assume you are doing one train-test split where you build the model on the training set and validate on the test set. If n was very large so that there would be trivial misspecification error even when using $K = 2$, would there be any benefit at all to increasing K if your objective was to estimate generalization error? Explain.
Not really, it can provide slightly more stable error estimates. But it will be too much computational power to already split it up more than when K=2, and you won't really get anything good out of it

(c) [easy] What problem does K-fold CV try to solve?

It ensures that each data point is used for testing and training

(d) [difficult] [MA] Theoretically, how does K-fold CV solve this problem? The Internet is your friend.