MATH 342W / 650.4 Spring 2024 Homework #3

Loyd Flores

Monday 18th March, 2024

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 in itself is very complex. There are too many reasons for why things occur, too many variables to consider, we can never completely know what goes on in a phenomena, the best we can do is approximate. Weather is a different beast. It is a dynamic system where the tiniest things could drastically alter the outcome. This is one reason why predicting the weather is hard. There exists limitation on how much we understand the world. Due to it's complexity it makes it even harder to collect the right data to input into our models. For example, even thermometers make mistakes. As stated in Silver's book, they may even be off just in the third or fourth decimal place. According to the principles of chaos theory and the nature of weather prediction as a dynamic system implies that fundamentally there is already error occurring and piling up. Our prediction from the start is immediately wrong. In the context of our lectures there exists such $z_1, z_2, \ldots, z_n$, which are real drivers of a phenomena. We may never even grasp the $z's$ that's why the best we can do is approximate them with $x's$, which again makes our predictions wrong. Returning back on the idea of Weather being a dynamic system, that in itself also poses another problem. The features we use to predict today may not be of significance in the future. We may find other reasons as to why the weather changes thus debunking methodologies we used in the past. Weather is not stationary and a multitude of factors may contribute to the change of weather and temperature. Lastly, despite computing power improving exponentially in recent decades it still isn't enough. Our ability to compute the weather has long lagged behind our theoretical understanding of it. We know which equations to solve and roughly what the answers are , but we aren't fast enough to calculate tehem for every molecule in the earth's atmosphere, again because the world is too complex.

(b) [easy] Why does the weatherman lie about the chance of rain? And where should you go if you want honest forecasts?

According to the findings of Eric Floehr, That statistical reality of accuracy isn't the governing paradigm when it comes to commercial weather forecasting. In simpler terms commercial weather providers have less incentive to give the most accurate predictions and trade-off some accuracy to fuel their personal agendas. Most commercial weather forecasts predict more precipitation than what actually occurs. Meteorologists call this "wet bias". For example if the prediction comes out to be a 5%, they instead inflate it to be 20%. By doing this commercial weather forecasts gain more profit because they "add value" by subtracting accuracy. According to Floehr if you want the most accurate forecasts you must result to the source, the government's data. All the commercial providers source their data from here before aggregating their own biases onto the data. The further you are from source the less accurate it becomes.

(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.

Both weather and earthquakes are dynamic systems. There are a multitude of factors that could affect these natural phenomena from occurring and a slight mishap in calculation could render our model to occur more error. The first major difference is that meteorologists are approximating the $z's$ more accurately. This roots from the better understanding of meteorologists on weather's phenomena or dynamic system as to seismologists to the phenomena of earthquakes. For example they have a strong fundamental understanding of what causes tornadoes and how they dissipate and seismologists are still trying to find out for sure what happens before or after an earthquake. Therefore models predicting earthquakes will incur larger δ, or error due to ignorance, there are just more things that seismologists are unaware of. A possible cause could be that weather has a richer dataset $\mathbb{D}$ available since earthquakes happen less frequently. A lot of earthquakes occur but they all vary in strength. Weaker ones happen more frequently and some even go unnoticed or unregistered while stronger ones happen so infrequently that we can't even begin to pick up the patterns. There is an obvious difference in the availability of data and understanding of phenomena. Secondly weather has a larger array of out of sample validation techniques that it can rely on. Persistence, climatology, and larger amounts of data is available for weather meanwhile earthquakes can only be validated once they occur. In the terms of our lecture, weather has better validation techniques and is even able to split its dataset $\mathbb{D}$ into D_{test} and D_{train} , while earthquakes don't really have the same luxury. Earthquake prediction desperately seeks signal while weather is gradually developing their approximations to f.

(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?

As discussed in Page 163 most economists rely on their judgement to some degree when making a forecast rather than just taking the output of a statistical model as is. Which could be simplified using the a quote from *George Box* which states: "All models are wrong but some are useful." It is then necessary that judgemental adjustments are made to the statistical model to make them better which was proven in the study of *Stephen K. McNess*, adjusted models performed 15% better because relying solely on statistical methods and computational resources will not account for the lack of theoretical understanding about a phenomena. What he was discussing as the nonsense predictor is when too much judgment is introduced it turns into bias. The amount of bias that could be incurred depends on the influence of reputation. The less known you are the less you have to lose by taking a big risk when predicting. Conversely if you have a good reputation you might be reluctant to step too far out of the line even if data demands it. Either of these concerns potentially distracts you from the goal of making the most honest predictions. The data collected from the Survey of Professional Forecasters proves that anonymous predictions always performed better because there are less biases and personal interests are added onto the model due to anonymity being lower stakes.

(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?

Over fitting is a big topic in this part of the text. It occurs when a model captures the noise rather than the underlying signal, resulting in a model that performs well on the training data but poorly on new, unseen data. In the context of our class an over-fit model understands the variability of the data it is fed which in the class' context means the model has an R^2 which is closer to one, in the context of this text it was 85% . Although these in-sample metrics are amazing they do not tell the entire truth. If anything they are deceiving you. The model fails to generalize new or unseen data. The quote **"With four parameters I can fit an elephant"** implies that the possibilities for an over fit model are endless even with a small feature count of 4. It emphasizes this even more by stating **"With five I can make him wiggle his trunk"** that the addition of more noise could drastically increase the complexity of our model. This implies that our model is so delusional that because it fully understood the noise and its complexity which places us further away from the actual phenomena we are interested in.

(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.

Predicting unemployment is different from weather or earthquakes because it has even less data available while being more complex. The nature of employment and economy is a different level of complex, there are so many indicators that you could possibly consider. It also requires a really complex model to understand the large variability in this field. In the context of the lectures the problem with unemployment predictions is that we are unable to properly identify the proper approximation x to the $z's$. There are so many interconnected variables from small consumer behavior to large global movements of the market. All these indicators are dynamic and we can't keep track of them which increases δ that our model is incurring or the error due to ignorance. Economic models also require a lot more subjective judgement allowing over-fitting wreak havoc easily. With the complex nature of the economy, it is harder to approximate f, increasing our misspecification error due to the limited nature of existing models. Overall the problem is just completely different in complexity. It stems from the unique and complex nature of economic systems, which pose challenges beyond those encountered in weather or earthquakes.

(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.

I agree with this statement. To solve a problem you must first understand the problem. For example you may memorize all the mathematical formulas in this world but if you are unable to understand the problem the knowledge you posses is rendered unusable due to the fact that you are unable to apply it. The same could be said in the context of Data Science and Machine Learning. How will you solve problems if you don't know how they fundamentally occur? The lack of theory presents a worse approximation of $x's$ to $z's$. There is also room for oversimplifying the problem which leads to a model that also fails to approximate f. For someone to produce a model that works and is useful they must first understand what they are trying to achieve, this is done by understanding the theories that govern the desired target. Secondly, theory in the domains of machine learning and data science are also necessary to be able to interpret and utilize the available data. What is a science textbook to a dogmatic and religious fundamentalist, nothing because it is virtually blasphemous garbage to them.

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 $[\boldsymbol{H}]$?

If matrix X is size $n * (p+1)$ then rank$[H] = p+1$ should also hold because the rank of H is purely determined by the number of linearly independent columns in X which is $p+1$. The dimension of the column space of X is dictated by n, thus rank$[H] = p+1$

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

$= HX = X$, this could also be written as :
$= HX = X(X^T X)^{-1} X^T$ X = X
$= HX = X * I =$ X

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

The previous problem indicates that the effect of the **orthogonal projection matrix** H on matrix X is equivalent to simply leaving X unchanged. It aligns with the fundamental understanding of orthogonal projection, where vectors projected onto a subspace remain unchanged if they already lie within that subspace.

(d) [difficult] Let $\boldsymbol{X}'$ be the matrix of $\boldsymbol{X}$ whose columns are in reverse order meaning that $\boldsymbol{X} = [\boldsymbol{1}_n \vdots \boldsymbol{x}_{\cdot 1} \vdots \ldots \vdots \boldsymbol{x}_{\cdot p}]$ and $\boldsymbol{X}' = [\boldsymbol{x}_{\cdot p} \vdots \ldots \vdots \boldsymbol{x}_{\cdot 1} \vdots \boldsymbol{1}_n]$. Show that the projection matrix that projects onto colsp $[X]$ is the same exact projection matrix that projects onto colsp $[X']$.

$$X = [1_n \vdots X_{\cdot 1} \vdots \cdots \vdots X_p]$$
$$X' = [X_p \vdots \cdots \vdots X_{\cdot 1} \vdots 1_n]$$

→ Given that H projects onto colspc[x], H' should projet to colspc[x']. Therefore we must prove that H = H'

$$H' = X'(X'^T X')^{-1} X'^T$$

$$= H' = ([x_p \cdots x_1 \cdot 1])\left(\left(\begin{bmatrix} x_p \\ \vdots \\ x_1 \\ 1n \end{bmatrix}\right)^T \left(\begin{bmatrix} x_p \\ \vdots \\ x_1 \\ 1n \end{bmatrix}\right)\right)^{-1} ([x_p .. x_1 \ 1n])^T$$

→ $X'^T X'$ has the same structure to $X^T X$ because reversing the order does not change the inner product between columns therefore we proved

$(x'^T x')^{-1} = (X^T X)^{-1}$ hence $\boxed{H' = H}$

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

For something to be an orthogonal projection matrix it must satisfy two conditions, **Symmetry** and **Impotence**. In the case of I_n by definition it is the identity matrix meaning that all diagonal elements are 1 and the rest are 0's. Since it is the identity it has to be square. Multiplying such matrix by itself you obtain yourself or $I_n^2 = I_n$ which satisfy the condition of **symmetry**. Another implication of I_n being square is that if transpose or swap the rows and columns it does not change the structure or $I_n^T = I_n$ satisfying the second condition of **idempotency**.

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

The identity matrix I_n projects onto the entire subspace of $\mathbb{R}^2$ because as the identity matrix it preserves all vectors in $\mathbb{R}^2$ without changing the direction or magnitude of any vector v. When we apply the projection operation using I_n we get the same vector back or I_nv = v.

(g) [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?

If rank$[X] = p+1$, its degrees of freedom would also be p+1. The Degrees of freedom just imply the number of linearly independent columns we have that can capture the variability in the response variable. In simpler terms we have more independent pieces of information to explain our target hence why "degrees of freedom capture variability."

(h) [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?

To project y onto the colspc$[X]$, the projection matrix could be denoted as $H = X(X^TX)^{-1}X^T$. Substituting that we get:
$= Proj_{colsp[x]}(y) = (X(X^TX)^{-1}X^T)y$

In OLS the goal is the find the coefficients β that minimize the residual sum of squares, β is denoted by :
$\beta = (X^TX)^{-1}X^T)y$

We can substitute the OLS solution by plugging in β :
$Proj_{colsp[x]}(y) = (X(X^TX)^{-1}X^T)y =$ Xβ. Therefore in the context of OLS regression, the projection of y into $colsp[x]$ is the same as the predicted values of y from the regression model.

(i) [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.

The second regression on the residuals may not necessarily result in $\hat{e}_{new}$ that is closer to 0_n than the initial residual e because the initial regression already captures the variation in y explained by the predictors in X. The remaining variation captured by the residuals may not be effectively modeled by the same predictors in X, since we are using the same input matrix X for the succeeding iterations. Thus implying that iterative regression on the residuals may not significantly improve the modal and may not converge to a better solution compared to iterative method that perception uses that it gradually fixes the fit of the line.

(j) [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.

Since Q is an **orthonormal matrix** its transpose Q^T is also its inverse. This is the fundamental idea of orthonormal matrices that they have orthogonal columns and unit length, when transposed they maintain their orthogonality and unit length thus proving $Q^{-1} = Q^T$

(k) [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.

1. Define H

   $H = X(X^TX)^{-1}X^T$

2. Substitute Q for H

   $H = Q(Q^TQ)^{-1}Q^T$

3. Q is orthogonal

   $Q^TQ = I$ (identity) therefore

   $H = Q(I)^{-1}Q^T$

4. The inverse of the identity is itself

   Since $I^{-1} = I$ we obtain

   $H = QIQ^T$

5. The Identity matrix does not affect it could be removed proving that

   $\boxed{H = QQ^T}$

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

The projection onto Q will be denoted as P_Q.

we want to prove that $P_Q = \sum_{i}^{p+1} q_i q_i^T$ where q_i is the i^{th} column in Q, which shows the sum of projections

→ $P_Q = Q(Q^TQ)^{-1}Q^T$, since Q is orthonormal ($Q^TQ = I$) we can simplify this to

$P_Q = QI^{-1}Q^T = QQ^T$

To prove they are equal:

Let v be an arbitrary vector

1) v onto P_Q: $P_Q v = (QQ^T)v$

2) v onto sum of projections: $\left(\sum_{i}^{p+1}(q_i q_i^T)v\right)$

After all iterations

$$\boxed{\sum_{i}^{p+1}(q_i q_i^T)v = QQ^T v}$$

(m) [easy] Explain why adding a new column to $\boldsymbol{X}$ results in no change in the SST remaining the same.

SST or Sum squared total is a measure of the total variability present in the dependent variable y without considering any predictors. This is crucial because it provides a baseline understanding of how much the observed values of y deviate from their mean. SST is derived from SST $= \sum_{i=1}^{n}(y_i - \bar{y})^2$ adding more X's or feature columns won't really affect the variability of y.

(n) [harder] Prove that adding a new column to $\boldsymbol{X}$ results in SSR increasing.

SSR or Sum squared Residual represents the sum of the squared difference between the predicted values of the dependent variable $\hat{y}$ and the mean of the dependent variable $\bar{y}$ or SSR $= \sum_{i=1}^{n}(\hat{y} - \bar{y})^2$. When we add a new column X_{new} to the predictor matrix X it means that we're introducing a new predictor or feature. When we fit a linear regression model with the new predictor included. When the model attempts to explain more of the variability in the dependent variable incorporating all the features in X including X_{new} the values of $\hat{y}$ will probably change and will be closer to 0 therefore changing the value of SSR because the larger difference from $\hat{y}$ - $\bar{y}$ will result into a larger SSR when squared.

(o) [harder] What is overfitting? Use what you learned in this problem to frame your answer.

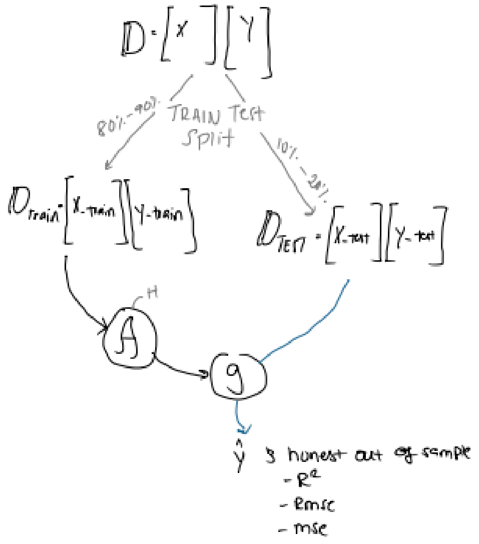
Overfitting occurs when our model oversimplifies a problem by fully understanding the patterns and noise found in the data it was trained on. It then fails to generalize on new and or unseen data. If I am using concepts from the previous questions, if we keep adding garbage features our model will become more complex and we will have a better fit line that includes the garbage. SSR will keep increasing which is a strong indicator of overfitting.

(p) [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.

In sample error metrics are dishonest because they are calculated using the same data the model was trained on. This means they can overly reward complex models that fit the noise in the data rather than the true pattern it needs. Better scores on in-simple metrics may lead you to think that you have an amazing model when in reality it won't really know how to deal with new or unseen data.

(q) [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.

To provide honest error metrics that accurately relfect the performance of a model, we need to verify it with data it has not seen. A model has to be validated with out-of-sample evaluation methods. One approach that we can do is split our $\mathcal{D}$ into $\mathcal{D}_{test}$ and $\mathcal{D}_{train}$, where $\mathcal{D}_{train}$ will be used to train our model and the remaining $\mathcal{D}_{test}$ will be used to validate our model using out-of-sample data.



(r) [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?

To produce more stable metrics we deploy a process called **K-fold cross validation**, which splits our data into **k** folds. For example we set $\mathbf{k} = 5$, this splits up the data into 5 folds. We then train 5 different models on these folds. Each model will then use one fold to validate and the rest to train. For example $model_1$ trains on $fold_{2..n}$ and uses $fold_1$ to validate. This repeats for all models and we average all the residuals to create a more stable metric.

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 a hyper parameter that could be selected, it signifies how you'll split the data. A low K means you'll have fewer data points in your test set, which could lead to higher bias in your evaluation because the model's performance could be sensitive to a particular subset of data for testing. A high K means you have more data points in your test set which could help reduce bias. However, this might increase variance because your averaging over more test sets leading to a wider range of performance metrics. To summarize, K controls the bias-variance trade off in models.

(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.

If n is very large and there is trivial misspecification error present a larger n may still contain the same misspecification error accounted for. Since there is a large n our model probably fit the noise as well. The only way to develop our model is by picking a more complex set of functions in our H. In the end when trivial misspecification is present there is no benefit in increasing K

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

K-fold cross-validation aims to address problems associated with the traditional train-test split. According to silver's book it is always better to provide a range of predictions rather than a single prediction by itself. This concept is sort of applicable in this scenario, rather than verifying your model with one error, you instead take the error of multiple models and average out providing a more stable out of sample metric. With a single train-test split the performance of the model can be slightly sensitive to particular subset of the data. If the split unluckily splits including extreme points the metric won't be so honest. This is solved by deploying k-fold cross-validation attempts to take a piece of the entire dataset and get the best and most stable metridcs.