Lecture 7: Regression II: Theory and Practice
Modeling Social Data, Spring 2017
Columbia University

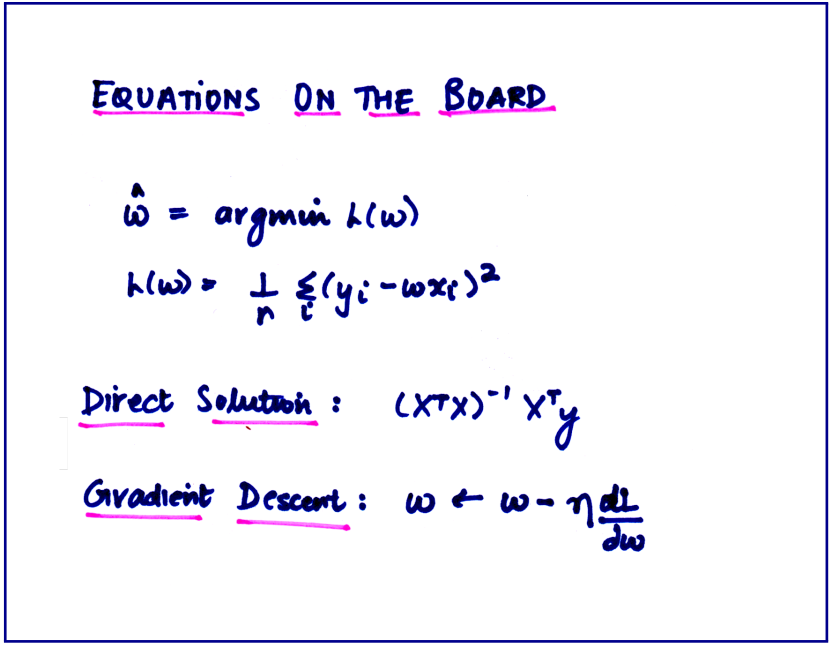
Aashima Arora

March 3, 2017

1 Bits from the last lecture

The least squares closed form solution is expensive. It scales in a way such that it would need cubic time ($\mathcal{O}(n^3)$) to perform the operations. On huge datasets, cubic times gets really bad. For example, suppose there are 100000 features, we would be processing a 100000 x 100000 matrix. So, *Gradient Descent* is another option. It should get us to the bottom as the closed form solution. Each one of the steps takes $n * k$ operations. It is computationally cheap though it has certain downsides. It is iterative and you need to be careful of selecting learning rate η. For really wide datasets, even $n * k$ operations are a lot. In this case, we opted for *Stochastic Gradient Descent* . In SGD, we approximate the gradient with a random sample. It helps us control how many points to look at before computing the gradient.

Figure 1: Least Square Solutions



EQUATIONS ON THE BOARD

$$\hat{w} = \text{argmin } L(w)$$

$$L(w) = \frac{1}{n} \sum_i (y_i - wx_i)^2$$

Direct Solution : $(X^T X)^{-1} X^T y$

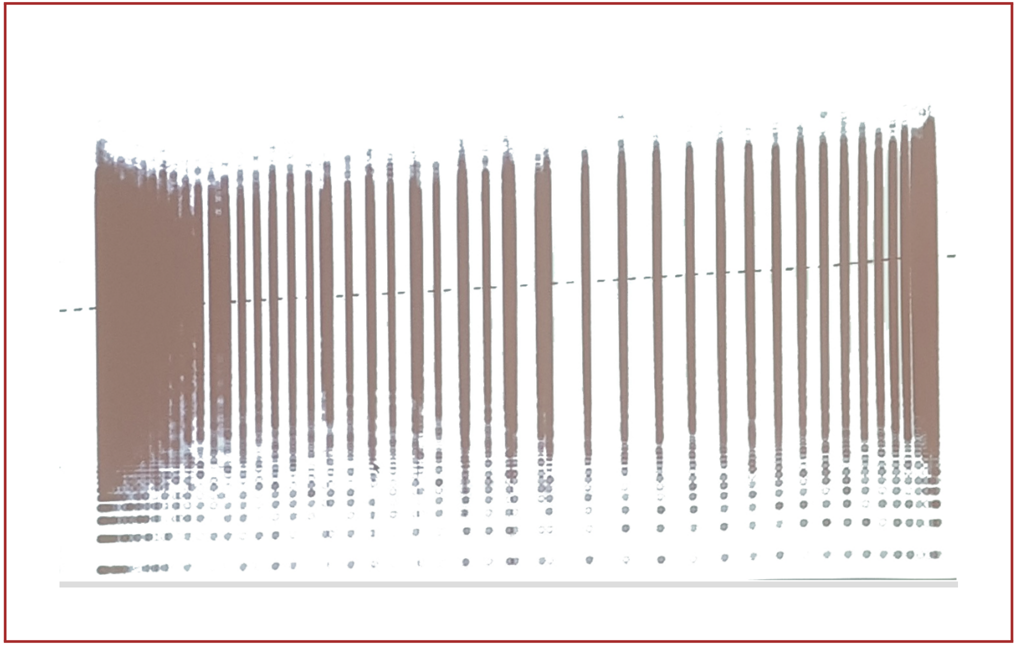
Gradient Descent : $w \leftarrow w - \eta \frac{dL}{dw}$

2 Model Selection and Evaluation

The question arises that *When should we stop fitting?* . Here, in the example where we were trying to model internet viewing using age and gender interactions, we knew that we had to fit a parabola. Here we already had the model and we were figuring out how to select the parameters. We did not discuss how to select the model. We usually find the best parameters and visually inspect the fit by plotting the predicted and the actual values.

We summarized the view as average viewing which is not the same as looking at all the ages. If we look at the genders of all the ages in the figure below, we observe that there exists a lot of individual variability conditioned on age. The variation between the age is more than the systematic individual variance. The average by age changes a tiny bit. Given that the average changed, doesn't mean every person changed. This is why it is not such a good model. We can't interpret the model just by staring at the coefficients. There is a lot that has not been explained. Sometimes, modeling the average is a good idea. If we are making an economic policy, we care about the average more than every individual. Whether we have achieved what we want, depends on the goal. We might want to do analysis **quantitatively** which would actually help us answer if the systematic shift in the mean would explain the variance in the world. Infact, it doesn't and we shall see how simply age and gender does not explain internet viewing!

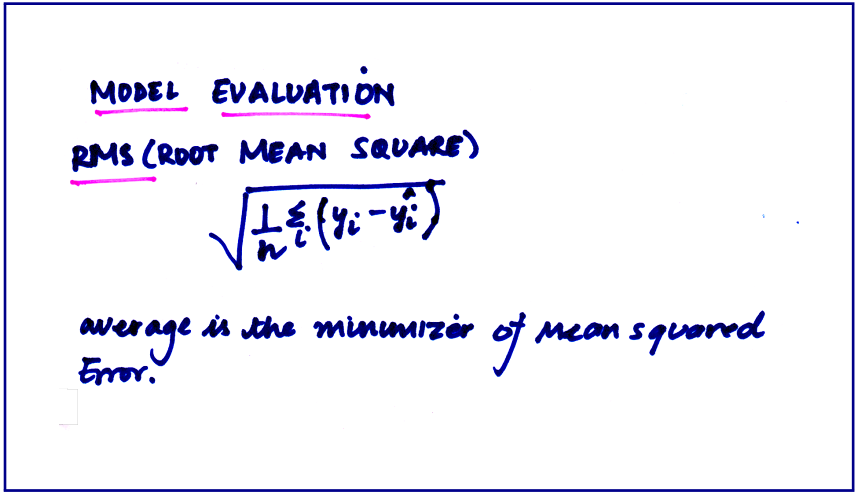
Figure 2: Internet viewing by gender of all ages



3 Performance metrics for model fitting

The most common metric for evaluating a model is *Root mean square error.* The square root in the formula is to enforce the same unit. Saying that we are 10000 square page views off doesn't make much sense.

Figure 3: RMS error



MODEL EVALUATION

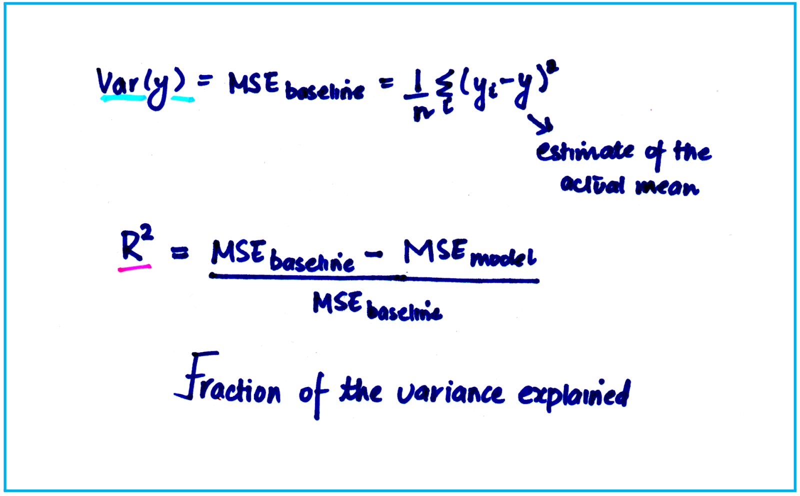
RMS (ROOT MEAN SQUARE)

$$\sqrt{\frac{1}{n}\sum_i (y_i - \hat{y}_i)}$$

average is the minimizer of mean squared Error.

The baseline is to predict the mean for each observation. For any regression problem, we usually measure the performance of the model with respect to the baseline model and ask how does it compare to the average. The following is the representation of the normalized measure we use to evaluate the model. This measure is called R^2.

Figure 4: R^2



$$Var(y) = MSE_{baseline} = \frac{1}{n}\sum_i (y_i - y)^2$$

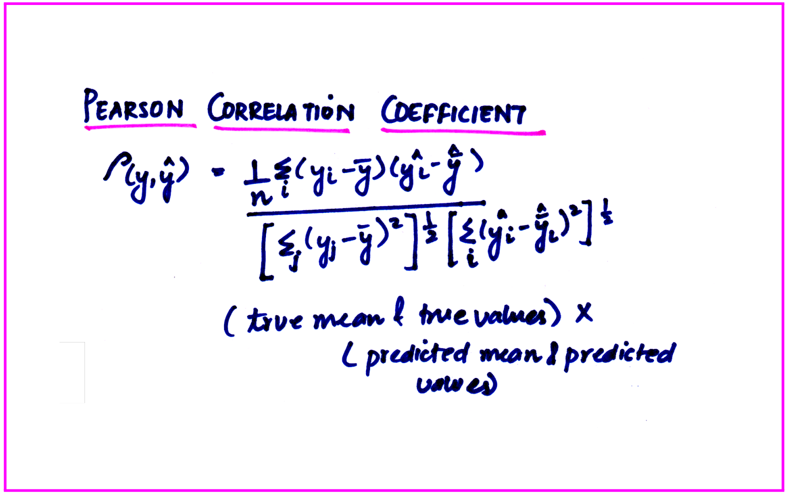
estimate of the actual mean

$$R^2 = \frac{MSE_{baseline} - MSE_{model}}{MSE_{baseline}}$$

Fraction of the variance explained

R^2 explains the fraction of variance in the model. The R value of the above model was 0.019 . It means that this model can capture only 2% of the variance of the whole dataset. Terrible! If R is 0, it is pretty bad while if it is 1, the model is great. This value is not unrelated to correlation that is represented by the Pearson's coefficient. The Pearson's coefficient shows how correlated are the deviations of the true and predicted values are, from the mean. We can use features like glance in R to look at the various coefficients. It is very well known that RMS is extremely sensitive to outliers and that big deviations contribute a lot to mean squared error.

Figure 5: Pearson's Correlation Coefficient



PEARSON CORRELATION COEFFICIENT

$$\rho(y,\hat{y}) = \frac{\frac{1}{n}\sum_i (y_i-\bar{y})(\hat{y}_i-\bar{\hat{y}})}{\left[\sum_j (y_j-\bar{y})^2\right]^{\frac{1}{2}}\left[\sum_i (\hat{y}_i-\bar{\hat{y}}_i)^2\right]^{\frac{1}{2}}}$$

(true mean & true values) x (predicted mean & predicted values)

4 The goal is not to get the training error down to 0!

While trying to explain the past in your model, you should be careful of not trying to focus on explaining it too well. The goal is to **generalize** i.e predict well on the unseen data and not try to fit a complex decision boundary that goes through all or most of the training data points. We can use the function $poly(age, k)$ and add more degrees to the decision boundary but this will lead to overfitting. In doing this in *R*, you might get an NA which means that your data points are extremely correlated such that *R* treats them as identical. The thing to remember is that you can do *worse* in the future if pay too much attention to the past.

4.1 Cross Validation

The dataset is usually split into three sets -

1. Training - This data is used to fit the model and find the best set of parameters.
2. Test - This should **not be touched** until we have tested our model on the validation set.
3. Validation - This data is used to test the performance of the training data before it actually exposed to the test set. We should be careful and should not overfit to the validation set.

A famous method for validation is **K-Fold Validation**. Here we select a value of K(generally 5 or 10). A training row is randomly selected and is added to the fold and we go through each fold. For each fold, the other K - 1 folds acts as the training data and the fold selected is treated as the test set. We then select the model with the lowest cross validation error.

5 Bias/Variance Tradeoff

5.1 Bias?

Bias is the true error of the best classifier in the concept class (e.g, best linear separator, best decision tree on a fixed number of nodes). Basically, you approximate a function $\hat{f}$ and you compare how much it is off from the true function *f*.

1. High Bias : If your model has high bias, it means that it assumes something that might not be true, so you could always be off even if you have an infinite amount of training data!
2. Low Bias : If your model has low bias, it means that it can capture complicated patterns with enough data. For example, if the world is really 9th degree, eventually you will find the right model.

5.2 Variance?

Variance is the error of the trained classifier with respect to the best classifier in the concept class. It is a measure of how much your estimate changes across the different datasets($\hat{f}_1$,$\hat{f}_2$,$\hat{f}_3$)

1. High Variance : If your model has high variance, it means your model changes a lot with different training datasets.

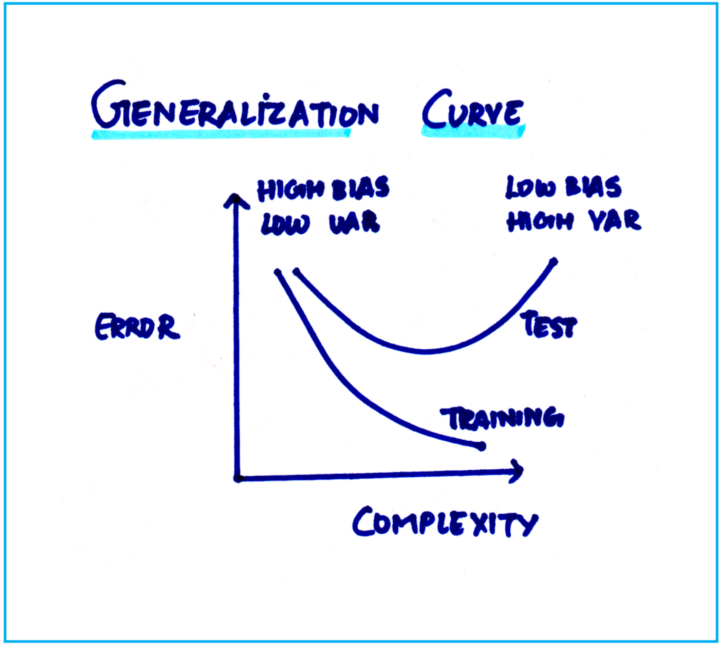
2. Low Variance : If your model has low variance, it means that the estimate doesn't change a lot with different datasets.

5.3 Tradeoff?

If we make the concept class more complicated (e.g, linear classification to quadratic classification, or increase number of nodes in the decision tree), then bias decreases but variance increases(Overfitting). On the other hand, if try to lower the variance, we are making the model too simplistic that it cannot capture the true distribution(Underfitting). Thus there is a bias-variance tradeoff.

The equation that captures the mean squared error is **MSE** $=$ **Bias**^{2} $+$ **Variance - Irreducible Error**

Figure 6: Generalization Curve

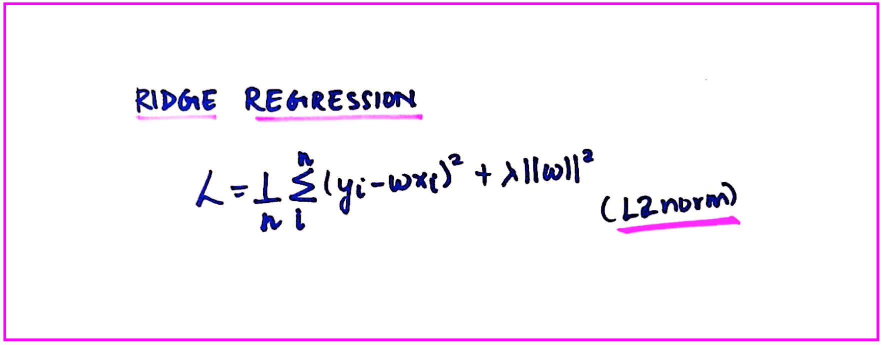


6 Regularization

Regularization artificially discourages complex or extreme explanations of the world even if they fit the what has been observed better. The idea is that such explanations are unlikely to generalize well to the future; they may happen to explain a few data points from the past well, but this may just be because of accidents of the sample. It, technically attempts to solve the overfitting problem in statistical models.

1. Ridge Regression - Ridge regression is a way of shrinking weights. It penalizes the loss function using a parameter λ times the sum of squared weights. It establishes that it doesn't believe in the weights too much and tries to bring them down in an attempt to generalize well.

Figure 7: Ridge Regression



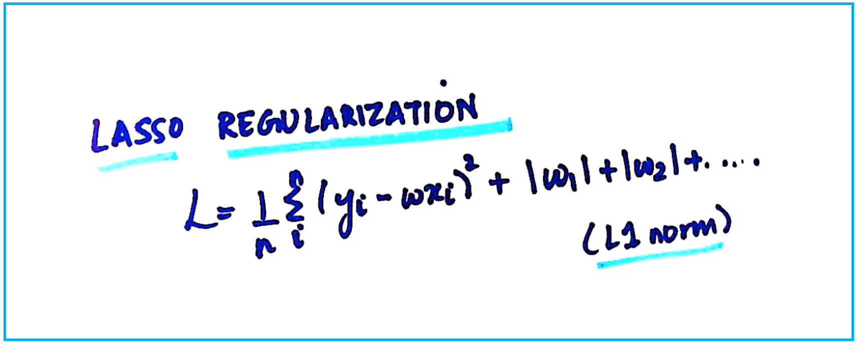
RIDGE REGRESSION

$$\mathcal{L} = \frac{1}{n}\sum_{i}^{n}(y_i - \omega x_i)^2 + \lambda\|\omega\|^2$$

(L2 norm)

2. Lasso Regression -Lasso is a way to perform feature selection. Here, instead of taking the squared norm,we take the absolute norm also known as $L1$ norm. Lasso ropes in a bunch of things and sets them explicitly to zero. It believes that if it had to choose then it would zero out a coefficient than make it small.

Figure 8: Lasso Regularization



LASSO REGULARIZATION

$$\mathcal{L} = \frac{1}{n}\sum_{i}^{n}(y_i - \omega x_i)^2 + |\omega_1| + |\omega_2| + \ldots.$$

(L1 norm)