Section 03: Solutions

Curated by. Leo Liu (zeyuliu2@cs.washington.edu) and Ethan Chau (echau18@cs.washington.edu).

1. Generalized Least Squares Regression

Here we consider a problem that generalizes both of these. As a reminder, in linear regression, we seek a model that captures a linear relationship between input data and output data. The general case we consider imposes additional structure on the model.

Consider an experiment in which you have n data points $x_i \in \mathbb{R}^d$ and corresponding n observations y_i. We wish to come up with a model $\omega \in \mathbb{R}^d$ that satisfies the following properties: first, the error $\sum_{i=1}^n (x_i^\top \omega - y_i)^2$ should be small; second, we don't want small changes in training data resulting in large changes in solution; third, we want to put different weights in controlling the magnitude of different coordinates of ω. We therefore define

$$\widehat{\omega}_{\text{general}} = \arg\min_{\omega} \sum_{i=1}^{n} (y_i - x_i^\top \omega)^2 + \lambda \sum_{i=1}^{d} D_{ii} \omega_i^2.$$

Here, D is a diagonal matrix, with positive entries on the diagonal. Observe that when D is the identity matrix, we recover ridge regression, and when $\lambda = 0$, we recover least squares regression. Different weights on D_{ii} cause the magnitudes of ω_i to be controlled differently.

1.1. Closed form in the general case

Deduce the closed form solution for $\widehat{\omega}_{\text{general}}$. You should be comfortable with proofs in the "coordinate" form as well as the "matrix" form.

Solution:

We first give the proof using "matrix" notation. The objective function can be expressed as

$$\begin{aligned} f(\omega) &= \|X\omega - y\|_2^2 + \lambda \omega^\top D \omega \\ &= (X\omega - y)^\top (X\omega - y) + \lambda \omega^\top D \omega \\ &= (X\omega)^\top X\omega - (X\omega)^\top y - y^\top X\omega + y^\top y + \lambda \omega^\top D \omega \\ &= \omega^\top X^\top X \omega - 2\omega^\top X^\top y + y^\top y + \lambda \omega^\top D \omega \\ &= \omega^\top (X^\top X + \lambda D)\omega - 2\omega^\top X^\top y + y^\top y \end{aligned}$$

The gradient of f is

$$\begin{aligned} \nabla f(\omega) &= \nabla_\omega (\omega^\top (X^\top X + \lambda D)\omega - 2\omega^\top X^\top y + y^\top y) \\ &= \nabla_\omega (\omega^\top (X^\top X + \lambda D)\omega) - 2\nabla_\omega (\omega^\top X^\top y) + \nabla_\omega (y^\top y) \\ &= 2(X^\top X + \lambda D)\omega - 2X^\top y \end{aligned}$$

Here note that $X^\top X + \lambda D$ is a symmetric matrix, which explains the factor 2 in the gradient term. Setting the gradient $\nabla f(\omega)$ to zero, we can conclude that

$$(X^\top X + \lambda D)\widehat{\omega}_{\text{general}} = X^\top y$$

If $X^\top X + \lambda D$ is full rank then we can get a unique solution:

$$\widehat{\omega}_{\text{general}} = (X^\top X + \lambda D)^{-1} X^\top y$$

Since D is already given to be a diagonal matrix with strictly positive entries on the diagonal, any strictly positive λ will make the matrix $X^\top X + \lambda D$ invertible.

Solution:

We now give a solution in the "coordinate" form. The objective, when written in coordinate form, is $f(\omega) = \sum_{i=1}^n (y_i - x_i^\top \omega)^2 + \lambda \sum_{i=1}^d D_{ii}\omega_i^2$. As in the previous proof, we first simplify it as follows and then set it zero:

$$\begin{aligned}
\nabla_\omega \left[\sum_{i=1}^n (y_i - x_i^\top \omega)^2 + \lambda \sum_{i=1}^d D_{ii}\omega_i^2 \right] &= \nabla_\omega \sum_{i=1}^n (y_i - x_i^\top \omega)^2 + \nabla_\omega \lambda \sum_{i=1}^d D_{ii}\omega_i^2 \\
&= \sum_{i=1}^n \nabla_\omega (y_i - x_i^\top \omega)^2 + 2\lambda D\omega \\
&= -\sum_{i=1}^n 2x_i(y_i - x_i^\top \omega) + 2\lambda D\omega \\
&= -\sum_{i=1}^n 2x_i y_i + \sum_{i=1}^n 2x_i x_i^\top \omega + 2\lambda D\omega \\
&= -2\sum_{i=1}^n x_i y_i + 2\left(\sum_{i=1}^n x_i x_i^\top + \lambda D \right) \omega \\
&= 0 \quad \text{(set it to be 0)}
\end{aligned}$$

$$\widehat{\omega}_{\text{general}} = \left(\sum_{i=1}^n x_i x_i^\top + \lambda D \right)^{-1} \left(\sum_{i=1}^n x_i y_i \right)$$

Note that, as expected, this exactly matches the answer we got from the previous approach (because x_i's are all the rows of X, and therefore $\sum_i x_i y_i = X^\top y$, and $\sum_i x_i x_i^\top = X^\top X$).

1.2. Special cases: linear regression and ridge regression

(a) In the simple least squares case ($\lambda = 0$ above), what happens to the resulting $\widehat{\omega}$ if we double all the values of y_i?

Solution:

As can be seen from the formula $\widehat{\omega} = (X^\top X)^{-1} X^\top y$, doubling y doubles ω as well. This makes sense intuitively as well because if the observations are scaled up, the model should also be.

(b) In the simple least squares case ($\lambda = 0$ above), what happens to the resulting $\widehat{\omega}$ if we double the data matrix $X \in \mathbb{R}^{n \times d}$?

Solution:

As can be seen from the formula $\widehat{\omega} = (X^\top X)^{-1} X^\top y$, doubling X halves ω. This also makes sense intuitively because the error we are trying to minimize is $\|X\omega - y\|_2^2$, and if the X has doubled, while y has remained unchanged, then ω must compensate for it by reducing by a factor of 2.

(c) Suppose $D = I$ (that is, it is the identity matrix). That is, this is the *ridge* regression setting. Explain why $\lambda > 0$ ensures a "well-conditioned" setting.

Solution:

The solution is $\widehat{\omega} = (X^\top X + \lambda I)^{-1} X^\top y$. We already saw in a previous part that $X^\top X$ is always positive semidefinite, that is, its eigenvalues are at least zero. Adding λI, where $\lambda > 0$, ensures that $X^\top X + \lambda I$ is in fact positive *definite*. This helps us have a good condition number.

2. Code reading

Is the test error unbiased for these programs? If not, how can we fix the code so it is?

2.1. Find the bug!

```
1  # Given dataset of 1000-by-50 feature
2  # matrix X, and 1000-by-1 labels vector
3
4  mu = np.mean(X, axis=0)
5  X = X - mu
6
7  idx = np.random.permutation(1000)
8  TRAIN = idx[0:900]
9  TEST = idx[900::]
10
11 ytrain = y[TRAIN]
12 Xtrain = X[TRAIN, :]
13
14 # solve for argmin_w ||Xtrain*w - ytrain||_2
15 w = np.linalg.solve(np.dot(Xtrain.T, Xtrain), np.dot(Xtrain.T, ytrain))
16
17 b = np.mean(ytrain)
18
19 ytest = y[TEST]
20 Xtest = X[TEST, :]
21
22 train_error = np.dot(np.dot(Xtrain, w)+b - ytrain,
23                 np.dot(Xtrain, w)+b - ytrain ) / len(TRAIN)
24 test_error = np.dot(np.dot(Xtest, w)+b - ytest,
25                 np.dot(Xtest, w)+b - ytest ) / len(TEST)
26
27 print('Train error = ', train_error)
28 print('Test error = ', test_error)
```

Solution:

The error is at the beginning of the program on lines 4 and 5. Notice how μ is a function of both the train and test data. By de-meaning the entire dataset before splitting, we are intertwining the train and test data. The correct procedure is:

- Split into train and test
- Compute the mean of the train data, μ_{train}
- De-mean both the train and test data with μ_{train}

2.2. K-fold Cross-Validation (Demonstrative code)

```
# Given dataset of 1000-by-50 feature
# matrix X, and 1000-by-1 labels vector
import np


def fit(Xin, Yin, lbda):
      mu = np.mean(Xin, axis=0)
      Xin = Xin - mu
      w = np.linalg.solve(np.dot(Xin.T, Xin) + lbda, np.dot(Xin.T, Yin))
      b = np.mean(Yin) - np.dot(w, mu)
      return w, b


def predict(w, b, Xin):
      return np.dot(Xin, w) + b


# Note: X, y are all the train data and label for the entire experiments
N_TRAIN = 1000
idx = np.random.permutation(N_TRAIN)
K_FOLD = 5

NON_TEST = idx[0: 9 * N_TRAIN // 10]
N_PER_FOLD = len(NON_TEST) // K_FOLD
TEST = idx[9 * N_TRAIN // 10::]

# candidates regularization coefficient to choose from
lbdas = [0.1, 0.2, 0.3]
err = np.zeros(len(lbdas))

for lbda_idx, lbda in enumerate(lbdas):
      for i in range(K_FOLD):
            # CRUCIAL: we use slicing to calculate the index train set and val set should use!
            # Using the ith fold as the validation set
            VAL = NON_TEST[i * N_PER_FOLD:(i+1) * N_PER_FOLD]
            # Using the rest as the train set
            TRAIN = NON_TEST[:i * N_PER_FOLD] + NON_TEST[(i + 1) * N_PER_FOLD:]

            ytrain = y[TRAIN]
            Xtrain = X[TRAIN]
            yval = y[VAL]
            Xval = X[VAL]

            w, b = fit(Xtrain, ytrain, lbda)
            yval_hat = predict(w, b, Xval)
            # accumulate error from this fold of validation set
            err[lbda_idx] += np.mean((yval_hat - yval)**2)

      # calculate the error for the k-fold validation
      err[lbda_idx] /= K_FOLD

lbda_best = lbdas[np.argmin(err)]

Xtot = np.concatenate((Xtrain, Xval), axis=0)
ytot = np.concatenate((ytrain, yval), axis=0)

w, b = fit(Xtot, ytot, lbda_best)

ytest = y[TEST]
```

```
    Xtest = X[TEST]

    ytot_hat = predict(w, b, Xtot)
    train_error = np.mean((ytot_hat - ytot) ** 2)
    ytest_hat = predict(w, b, Xtest)
    test_error = np.mean((ytest_hat - ytest) ** 2)

    print('Best choice of lambda = ', lbda_best)
    print('Train error = ', train_error)
    print('Test error = ', test_error)
```

3. Extra: Maximum *a posteriori* Estimation and Regularization

In the previous week's section, we demonstrated that under the linear measurement model $\forall i \in [n]. \, y_i = x_i^\top w + v_i$ with Gaussian noise ($v_i \sim \mathcal{N}(0, \sigma^2)$, the maximum likelihood estimator w is the minimizer of the square loss, i.e.,

$$\arg\max_w p_w((x_1, y_1), \ldots, (x_n, y_n)) = \arg\min_w \|Xw - Y\|_2^2.$$

Now, we will extend this result to regularization. Recall the following canonical probability distributions:

- Prior (model): $p(w)$
- Likelihood (data, given model): $p(Y \mid w, X)$
- Posterior (model, given data): $p(w \mid Y, X)$

In the above, X is in the condition because it is given as input. Maximum likelihood estimation derives its name from the fact that we maximize the likelihood distribution. We will now introduce the concept of maximum *a posteriori* estimation (MAP), which, as the name suggests, involves maximizing the posterior distribution. Note that:

$$\begin{aligned} p(w \mid Y, X) &= \frac{p(Y \mid w, X)p(w)p(X)}{p(Y \mid X)} \\ &\propto p(Y \mid w, X)p(w), \end{aligned}$$

i.e., maximizing the posterior is equivalent to maximizing the likelihood, weighted by some assumption on w (the prior). In this question, we will show that assuming certain priors on w corresponds to imposing certain types of regularization on the linear regression objective.

3.1. Preliminaries

(a) Derive an expression for the likelihood as a Gaussian distribution (hint: write the linear model in vector form and use properties of Gaussians).

Solution:

Let $v \in \mathbb{R}^n$ be the vector of all v_i, and notice that v is distributed as an isotropic Gaussian: $v \sim \mathcal{N}(0, \sigma^2 \mathbb{I})$. Then, we have $Y = Xw + v$. Since X and w are fixed in the likelihood term, we can treat the expression for Y as a linear transformation of a Gaussian, recovering the distribution:

$$Y \mid (w, X) \sim \mathcal{N}(Xw, \sigma^2 \mathbb{I}).$$

3.2. MAP as Regularization

(a) Suppose the elements of w are independently distributed according to a Laplacian distribution:

$$p(w_i) = \frac{\lambda}{4\sigma^2} \exp(-|w_i| \frac{\lambda}{2\sigma^2}).$$

Show that under this prior on w, MAP estimation of the linear measurement model recovers the LASSO objective.

Solution:

We work in the argmax space, which allows us to drop and add constants or monotonically increasing

functions as necessary.

$$\begin{aligned}
\arg\max_w p(w \mid Y, X) &= \arg\max_w \log p(w \mid Y, X) \\
&= \arg\max_w \log p(Y \mid w, X) + \log p(w) \\
&\overset{*}{=} \log \left\{ (2\pi)^{-\frac{n}{2}} \left(\sigma^2\right)^{-\frac{n}{2}} \exp(-\frac{1}{2}(Y - Xw)^\top (\sigma^2 \mathbb{I})^{-1}(Y - Xw) \right\} \\
&\quad + \sum_{i=1}^{n} \log \frac{\lambda}{4\sigma^2} \exp(-|w_i| \frac{\lambda}{2\sigma^2}) \\
&= \arg\max_w -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(Y - Xw)^\top (Y - Xw) \\
&\quad + n \log \frac{\lambda}{4\sigma^2} - \frac{\lambda}{2\sigma^2} \sum_{i=1}^{n} |w_i| \\
&\overset{**}{=} \arg\max_w -\frac{1}{2\sigma^2} \left[(Y - Xw)^\top (Y - Xw) + \lambda \sum_{i=1}^{n} |w_i| \right] \\
&\overset{***}{=} \arg\max_w -\|Y - Xw\|_2^2 + \lambda \|w\|_1 \\
&= \arg\min_w \|Y - Xw\|_2^2 + \lambda \|w\|_1,
\end{aligned}$$

where the first starred equality follows from applying the PDFs, and the second and third follow from dropping constant terms or multipliers. In other words, solving MAP with a Laplacian prior also solves the LASSO regression problem.

(b) Derive an expression for the prior on w that corresponds to the ridge regression objective. What is the significance of this result?

Solution:

Our high-level approach to this problem is to expand the terms of the objective to a form that resembles the core of a PDF, then attach the additive/multiplicative constants necessary to recover the full form of the PDF. We do this first with the likelihood term (since we know its form), and then with the prior.

$$\begin{aligned}
\arg\min_w \|Y - Xw\|_2^2 + \lambda \|w\|_2^2 &= \arg\max_w -(Y - Xw)^\top (Y - Xw) + -\lambda w^\top w \\
&= \arg\max_w -\frac{1}{2\sigma^2}(Y - Xw)^\top (Y - Xw) - \frac{\lambda}{2\sigma^2} w^\top w \\
&= \arg\max_w -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(Y - Xw)^\top (Y - Xw) - \frac{\lambda}{2\sigma^2} w^\top w \\
&= \arg\max_w \mathcal{N}(Y; Xw, \sigma^2 \mathbb{I}) - \frac{\lambda}{2\sigma^2} w^\top w \\
&= \arg\max_w \mathcal{N}(Y; Xw, \sigma^2 \mathbb{I}) - \frac{n}{2}\log\left(2\pi\frac{\sigma^2}{\lambda}\right) - \frac{1}{2} w^\top \left(\frac{\sigma^2}{\lambda}\mathbb{I}\right)^{-1} w \\
&= \arg\max_w \mathcal{N}(Y; Xw, \sigma^2 \mathbb{I}) + \mathcal{N}\left(w; 0, \frac{\sigma^2}{\lambda}\mathbb{I}\right).
\end{aligned}$$

In other words, our prior is that $w \sim \mathcal{N}\left(0, \frac{\sigma^2}{\lambda}\mathbb{I}\right)$.

This means that when we apply ℓ_2 regularization to our linear regression problem (i.e., ridge regression), we make the implicit assumption that our weight vector is drawn from a Gaussian prior. More generally, we can see that applying various forms of regularization correspond to different prior assumptions on w.

4. Extra: Understanding Stein's Paradox through bias-variance trade-off

In this problem, we'll use bias-variance tradeoff to find a non-obvious way of estimating the mean of unrelated distributions.

So far in class, we've always been trying to learn a function – given a bunch of features, understand how they predict the single-number output. In this problem, we're trying to do something a little different. We have n completely unrelated probability distributions. We're going to get one sample from each of the distributions, and attempt to predict each of their means. For some examples, our distributions might be: high temperature in Chicago on January 1st, low temperature in Seattle on December 1st, and your friend's score on the midterm.

More formally, let $\theta \in \mathbb{R}^n$ be the (unknown) true means of our n distributions. We will get a vector X where each $X_i \sim \mathcal{N}(\theta_i, \sigma^2)$. We're assuming that every distribution has the same variance, but our means could be very different. Our job is to report $\hat{\theta}$ to minimize our expected error: $\mathbb{E}\left[||\hat{\theta} - \theta||_2^2\right]$.

4.1. The Maximum Likelihood Estimator

The most natural estimator is the maximum likelihood estimator $\hat{\theta} = X$. It's not obvious that any other viable strategy exists. We'll use bias-variance tradeoff to show that there's actually a better estimator.

(a) Split the error into bias^2 and variance. I.e. show

$$\mathbb{E}[||\hat{\theta} - \theta||_2^2] = ||\mathbb{E}[\hat{\theta}] - \theta||_2^2 + \mathbb{E}[||\hat{\theta} - \mathbb{E}[\hat{\theta}]||_2^2]$$

Hint: add and subtract $\mathbb{E}[\hat{\theta}]$.

Solution:

We'll show two versions of this calculation. They're identical, but in one we use summations, and in the other we use vector notation.

$$\begin{aligned}
\mathbb{E}[||\hat{\theta} - \theta||_2^2] &= \mathbb{E}\left[\sum_{i=1}^n (\hat{\theta}_i - \theta_i)^2\right] = \sum_{i=1}^n \mathbb{E}\left[(\hat{\theta}_i - \theta_i)^2\right] \\
&= \sum_{i=1}^n \mathbb{E}\left[(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i] + \mathbb{E}[\hat{\theta}_i] - \theta_i)^2\right] \\
&= \sum_{i=1}^n \mathbb{E}[(\mathbb{E}[\hat{\theta}_i] - \theta_i)^2] + \sum_{i=1}^n \mathbb{E}[(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i])^2] + \sum_{i=1}^n \mathbb{E}\left[2(\mathbb{E}[\hat{\theta}_i] - \theta_i)(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i])\right] \\
&= \sum_{i=1}^n \mathbb{E}\left[(\mathbb{E}[\hat{\theta}_i] - \theta_i)^2\right] + \sum_{i=1}^n \mathbb{E}\left[(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i])^2\right] + \sum_{i=1}^n \mathbb{E}[2(\mathbb{E}[\hat{\theta}_i] - \theta_i)(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i])] \\
&= \mathbb{E}||\mathbb{E}[\hat{\theta}] - \theta||_2^2 + \mathbb{E}||\hat{\theta} - \mathbb{E}[\hat{\theta}]||_2^2 + \mathbb{E}[2(\mathbb{E}[\hat{\theta}_i] - \theta_i)(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i])]
\end{aligned}$$

Where we used linearity of expectation repeatedly. It is now enough to show that the final term is equal to 0. Indeed, note that $\mathbb{E}[\hat{\theta}_i] - \theta_i$ is just a number, so we can move the expectation inside to get the last term is: $2(\mathbb{E}[\hat{\theta}_i] - \theta_i) \cdot \mathbb{E}\left[(\hat{\theta}_i - \mathbb{E}[\hat{\theta}_i]\right]$ By linearity of expectation, the last factor in the product is 0 Thus the whole term is 0 as required.

Using vector notation:

$$\begin{aligned}\mathbb{E}\left[||\hat{\theta}-\theta||_2^2\right] &= \mathbb{E}\left[||\hat{\theta}-\mathbb{E}[\hat{\theta}]+\mathbb{E}[\hat{\theta}]-\theta||_2^2\right] \\ &= \mathbb{E}\left[||\hat{\theta}-\mathbb{E}[\hat{\theta}]||_2^2\right]+2\mathbb{E}\left[(\hat{\theta}-\mathbb{E}[\hat{\theta}])^T(\mathbb{E}[\hat{\theta}]-\theta)\right]+||\mathbb{E}[\hat{\theta}]-\theta||_2^2 \\ &= \mathbb{E}\left[||\hat{\theta}-\mathbb{E}[\hat{\theta}]||_2^2\right]+||\mathbb{E}[\hat{\theta}]-\theta||_2^2\end{aligned}$$

Where the last step is a result of $\mathbb{E}\left[\hat{\theta}-\mathbb{E}[\hat{\theta}]\right] = \mathbb{E}[\hat{\theta}]-\mathbb{E}[\mathbb{E}[\hat{\theta}]] = 0$.

(b) What is the variance of the estimator $\hat{\theta} = X$? Hint: Remember that for a random variable Z, $\text{Var}(Z) = \mathbb{E}[(Z-\mathbb{E}[Z])^2]$

Solution:

$$\begin{aligned}\mathbb{E}[||\hat{\theta}-\mathbb{E}[\hat{\theta}]||_2^2] &= \mathbb{E}\left[\sum_{i=1}^{n}\left(\hat{\theta}_i-\mathbb{E}[\hat{\theta}_i]\right)^2\right] \\ &= \sum_{i=1}^{n}\mathbb{E}\left[\left(\hat{\theta}_i-\mathbb{E}[\hat{\theta}_i]\right)^2\right] \\ &= \sum_{i=1}^{n}(\sigma^2) \\ &= n\sigma^2\end{aligned}$$

Where the third line follows from knowing that $\text{Var}(\hat{\theta}_i) = \sigma^2$.

(c) What is the bias^2 of the estimator $\hat{\theta} = X$?
Solution:

$$\begin{aligned}||\mathbb{E}[\hat{\theta}]-\theta||_2^2 &= \sum_{i=1}^{n}(\mathbb{E}[\hat{\theta}_i]-\theta_i)(\mathbb{E}[\hat{\theta}_i]-\theta_i]) \\ &= \sum_{i=1}^{n}(\theta_i-\theta_i)(\theta_i-\theta_i) \\ &= 0\end{aligned}$$

4.2. A Biased Estimator

The maximum likelihood estimator above is an unbiased estimator. However, if our goal is to minimize the expected mean squared error, can we sacrifice some bias to lower variance dramatically. Here's an estimator to consider: we shrink/bias the MLE estimate towards the mean of all observations, $\overline{X} = \frac{1}{n}\sum_{i=1}^{n}X_i$. To be exact, we would like to consider the estimator $\hat{\theta} = (1-\lambda)X + \lambda\overline{X}\mathbf{1}$ for some $0 \leq \lambda \leq 1$ where $\mathbf{1}$ is a vector will all 1s. We will show that for any θ, there will always be a such estimator (by choosing a proper λ) that better minimizes the mean squared error than the MLE. But this should sound crazy in two ways:

- We're intentionally guessing something we *know* is a biased estimator;
- **More importantly, now when we estimate the value $\hat{\theta}$, it becomes $(1-\lambda)X + \lambda\overline{X}\mathbf{1}$, where $\overline{X}$ depends on all observations $X_1, ..., X_n$.** Remember $X_1, ..., X_n$ are independent random variables.

(a) What is the variance of the estimator $\hat{\theta} = (1-\lambda)X + \lambda\overline{X}\mathbf{1}$?

Solution:

We'll first do an intermediate calculation that we'll need later:

$$\begin{aligned}
\mathbb{E}[(\overline{X}-\overline{\theta})^2] =& \frac{1}{n^2}\mathbb{E}\left[\left(\sum_{i=1}^{n}(X_i-\theta_i)\right)^2\right] \\
=& \frac{1}{n^2}\mathbb{E}\left[\sum_{i=1}^{n}\sum_{j=1}^{n}(X_i-\theta_i)(X_j-\theta_j)\right] \\
=& \frac{1}{n^2}\mathbb{E}\left[\sum_{i=1}^{n}(X_i-\theta_i)^2\right] \\
=& \frac{1}{n^2}\sum_{i=1}^{n}\sigma^2 \\
=& \frac{\sigma^2}{n}
\end{aligned}$$

Now let's calculate the variance.

$$\begin{aligned}
\mathbb{E}\left[||\hat{\theta}-\mathbb{E}[\hat{\theta}]||_2^2\right] &= \mathbb{E}\left[||(1-\lambda)(X-\theta)+\lambda(\overline{X}-\overline{\theta})\mathbf{1}||_2^2\right] \\
&= (1-\lambda)^2\mathbb{E}\left[||X-\theta||_2^2\right] + 2\lambda(1-\lambda)\mathbb{E}\left[(X-\theta)^T\mathbf{1}(\overline{X}-\overline{\theta})\right] + \lambda^2 n\mathbb{E}\left[(\overline{X}-\overline{\theta})^2\right] \\
&= (1-\lambda)^2 n\sigma^2 + 2\lambda(1-\lambda)n\mathbb{E}\left[(\overline{X}-\overline{\theta})^2\right] + \lambda^2 n\mathbb{E}\left[(\overline{X}-\overline{\theta})^2\right] \\
&= (1-\lambda)^2 n\sigma^2 + 2\lambda(1-\lambda)\sigma^2 + \lambda^2\sigma^2 \\
&= (1-\lambda)^2 n\sigma^2 + (1-\lambda+\lambda)^2\sigma^2 - (1-\lambda)^2\sigma^2 \\
&= (1-\lambda)^2(n-1)\sigma^2 + \sigma^2
\end{aligned}$$

(b) What is the bias^2 of the estimator?

Solution:

$$\begin{aligned}
||\mathbb{E}[\hat{\theta}]-\theta||_2^2 &= ||(1-\lambda)\theta + \lambda\overline{\theta}\mathbf{1} - \theta||_2^2 \\
&= \lambda^2||\overline{\theta}\mathbf{1}-\theta||_2^2 \\
&= \lambda^2\sum_{i=1}^{n}(\theta_i-\overline{\theta})^2
\end{aligned}$$

(c) What value of λ will result in the best estimator?

Solution:

As always, our strategy is to take a derivative and set it equal to 0. Our objective is $(1-\lambda)^2(n-1)\sigma^2 + \sigma^2 + \lambda^2\sum_{i=1}^{n}(\theta_i-\overline{\theta})^2$, so we want to solve

$$-2(1-\lambda)(n-1)\sigma^2 + 2\lambda\sum_{i=1}^{n}(\theta_i-\overline{\theta})^2 = 0$$

Solving for λ gives

$$\lambda = \frac{\sigma^2}{\sigma^2 + \frac{1}{n-1}\sum_{i=1}^{n}(\theta_i - \overline{\theta})^2}$$

(d) Compare the error you get from this biased estimator with unbiased estimator you have from section 2.1. Which one has smaller error?

Solution:

Let $S = \sum_{i=1}^{n}(\theta_i - \overline{\theta})^2 \geq 0$, we then have $\lambda = \frac{(n-1)\sigma^2}{(n-1)\sigma^2+S}$. Now we bound the overall mean squared error:

$$\begin{aligned}
&\lambda^2 \sum_{i=1}^{n}(\theta_i - \overline{\theta})^2 + (1-\lambda)^2(n-1)\sigma^2 + \sigma^2 \\
&= \left(\frac{(n-1)\sigma^2}{(n-1)\sigma^2 + S}\right)^2 S + \left(\frac{S}{(n-1)\sigma^2 + S}\right)^2 (n-1)\sigma^2 + \sigma^2 \\
&= (n-1)\sigma^2 \cdot S \cdot \left[\frac{(n-1)\sigma^2}{((n-1)\sigma^2 + S)^2} + \frac{S}{((n-1)\sigma^2 + S)^2}\right] + \sigma^2 \\
&= (n-1)\sigma^2 \cdot S \cdot \frac{1}{(n-1)\sigma^2 + S} + \sigma^2 \\
&\leq (n-1)\sigma^2 \cdot S \cdot \frac{1}{S} + \sigma^2 = n\sigma^2
\end{aligned}$$

The inequality is strict above when $\sigma^2 \neq 0$.

This is what Stein's paradox is pointing at — **we might need to intentionally inject bias, to reduce the overall error.** Maybe the injected bias uses seemingly irrelevant information (e.g. other independent variables).

4.3. James-Stein Estimator

The Bias-Variance Tradeoff says that since our error is just the sum of the bias^2 and the variance, if we can find a way to "tradeoff" bias for variance, we can affect our error. With our previous estimator, the two sources of error are quite imbalanced. None of our error is from bias, it all comes from variance. Can we think of a way to reduce variance (even if it means increasing the bias)?

Normally, the way we would reduce variance would be to sample the random variables again and take the average of the samples. But we can't do that for this problem (it would take us a whole year to get another high temperature on January 1st). Another way to decrease the variance is to "scale down" the random variable.

For example, in this section we are going to use the "James-Stein Estimator", $\hat{\theta}^{JS} = \left(1 - \frac{(n-2)}{||X||_2^2}\right) X$, where we scale our estimate by a factor of $1 - \frac{(n-2)}{||X||_2^2}$ less than 1.

It has been shown in [1], for any θ, the expected mean squared error of $\hat{\theta}^{JS}$ is consistently better than $\hat{\theta}^{MLE}$'s, i.e.,

$$\mathbb{E}[||\hat{\theta}^{JS} - \theta||_2^2] \leq \mathbb{E}[||\hat{\theta}^{MLE} - \theta||_2^2], \qquad \forall \theta \in \mathbb{R}^n$$

where $\hat{\theta}^{MLE} = X$ is the maximum likelihood estimator from 2.1. **Our estimator doesn't even depend on obtaining the optimal hyperparamter λ (as in 2.2) anymore.**

To understand the paradox, here are some essential ingredients and outline:

- The squared error is the sum of n individual errors — $\mathbb{E}[(\theta_1 - \hat{\theta}_1)^2], \mathbb{E}[(\theta_2 - \hat{\theta}_2)^2], \ldots$. In this case, the errors are calculated by squaring deviation from the true mean.

- When we do the scaling proposed by James and Stein, some of these individual errors will decrease, while some will increase.
- However, notice the function $(\cdot)^2$ will penalize the higher deviation more harshly than the smaller deviation, even though the total deviation remains the same.
- For example, $\mathbb{E}[(\theta_1 - \hat{\theta}_1)^2]$ could change from 100^2 to 90^2 while $\mathbb{E}[(\theta_2 - \hat{\theta}_2)]$ changes from 0 to 10^2, the overall squared error will decrease by $(100^2 - 90^2) - (0^2 - 10^2) = 1906$!

Now that we know how $\hat{\theta}^{JS}$ is trying to improve the overall squared error, we see that it is the objective $\min_{\hat{\theta}} \mathbb{E}[||\hat{\theta} - \theta||_2^2] = \sum_{i=1}^{n} \mathbb{E}[(\hat{\theta}_i - \theta_i)^2]$ that is dependent on all of the random variables $X_1, ..., X_n$. And to minimize this overall/joint objective, a good estimator of θ_i necessarily becomes dependent with other X_j's where $j \neq i$.

References

[1] *James, William and Stein, Charles*. [*Estimation with quadratic loss*]. Breakthroughs in statistics (Springer), 443–460, 1992.