Matheus C. Fernandes
fernandes@g.harvard.edu
CS181-S17

Assignment #1
Due: 5:00pm February 3, 2017

Collaborators:

Homework 1: Linear Regression

Introduction

This homework is on different forms of linear regression and focuses on loss functions, optimizers, and regularization. Linear regression will be one of the few models that we see that has an analytical solution. These problems focus on deriving these solutions and exploring their properties.

If you find that you are having trouble with the first couple problems, we recommend going over the fundamentals of linear algebra and matrix calculus. We also encourage you to first read the Bishop textbook, particularly: Section 2.3 (Properties of Gaussian Distributions), Section 3.1 (Linear Basis Regression), and Section 3.3 (Bayesian Linear Regression). (Note that our notation is slightly different but the underlying mathematics remains the same :).

Please type your solutions after the corresponding problems using this LaTeX template, and start each problem on a new page.

Problem 1 (Centering and Ridge Regression, 7pts)

Consider a data set $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ in which each input vector $\mathbf{x} \in \mathbb{R}^m$. As we saw in lecture, this data set can be written using the design matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ and the target vector $\mathbf{y} \in \mathbb{R}^n$.

For this problem assume that the input matrix is centered, that is the data has been preprocessed such that $\frac{1}{n}\sum_{i=1}^n x_{ij} = 0$. Additionally we will use a positive regularization constant $\lambda > 0$ to add a ridge regression term.

In particular we consider a ridge regression loss function of the following form,

$$\mathcal{L}(\mathbf{w}, w_0) = (\mathbf{y} - \mathbf{X}\mathbf{w} - w_0\mathbf{1})^\top(\mathbf{y} - \mathbf{X}\mathbf{w} - w_0\mathbf{1}) + \lambda \mathbf{w}^\top \mathbf{w}.$$

Note that we are not incorporating the bias $w_0 \in \mathbb{R}$ into the weight parameter $\mathbf{w} \in \mathbb{R}^m$. For this problem the notation $\mathbf{1}$ indicates a vector of all 1's, in this case in implied to be in $\mathbb{R}^n$.

(a) Compute the gradient of $\mathcal{L}(\mathbf{w}, w_0)$ with respect to w_0. Simplify as much as you can for full credit.

(b) Compute the gradient of $\mathcal{L}(\mathbf{w}, w_0)$ with respect to $\mathbf{w}$. Simplify as much as you can for full credit. Make sure to give your answer in vector form.

(c) Suppose that $\lambda > 0$. Knowing that $\mathcal{L}$ is a convex function of its arguments, conclude that a global optimizer of $\mathcal{L}(\mathbf{w}, w_0)$ is

$$w_0 = \frac{1}{n}\sum_{i=1}^n y_i \tag{1}$$

$$\mathbf{w} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y} \tag{2}$$

(d) In order to take the inverse in the previous question, the matrix $(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})$ must be invertible. One way to ensure invertibility is by showing that a matrix is *positive definite*, i.e. it has all positive eigenvalues. Given that $\mathbf{X}^\top\mathbf{X}$ is positive *semi*-definite, i.e. all non-negative eigenvalues, prove that the full matrix is invertible.

(e) What difference does the last problem highlight between standard least-squares regression versus ridge regression?

Solution

We can expand the equation:

$$\mathcal{L}(\mathbf{w}, w_0) = (\mathbf{y} - \mathbf{X}\mathbf{w} - w_0\mathbf{1})^\top(\mathbf{y} - \mathbf{X}\mathbf{w} - w_0\mathbf{1}) + \lambda \mathbf{w}^\top \mathbf{w}$$

into

$$\mathcal{L}(\mathbf{w}, w_0) = \mathbf{y}^\top\mathbf{y} - 2\mathbf{y}^\top\mathbf{X}\mathbf{w} - 2w_0\mathbf{y}^\top\mathbf{1} + 2w_0\mathbf{w}^\top\mathbf{X}^\top\mathbf{1} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} + w_0^2\mathbf{1}^\top\mathbf{1} + \lambda\mathbf{w}^\top\mathbf{w}.$$

Where re-arranging we obtain:

$$\mathcal{L}(\mathbf{w}, w_0) = \mathbf{y}^\top\mathbf{y} - 2\mathbf{y}^\top\mathbf{X}\mathbf{w} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} + \lambda\mathbf{w}^\top\mathbf{w} + w_0^2\mathbf{1}^\top\mathbf{1} - 2w_0(\mathbf{y}^\top\mathbf{1} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{1}).$$

(a)

Taking the partial w.r.t. our expanded equation above we obtain the following

$$\frac{\partial \mathcal{L}(\mathbf{w}, w_0)}{\partial w_0} = 2w_0\mathbf{1}^\top\mathbf{1} - 2\left(\mathbf{y}^\top\mathbf{1} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{1}\right)$$

where $\mathbf{1}^\top\mathbf{1} = n$, which then becomes

$$\frac{\partial\mathcal{L}(\mathbf{w}, w_0)}{\partial w_0} = 2w_0 n - 2\left(\mathbf{y}^\top\mathbf{1} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{1}\right)$$

Because the above stated assumption in that the data has been centered (by removing the mean) $\frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0$, this implies that $\mathbf{w}^\top\mathbf{X}^\top\mathbf{1} \to 0$, such that

$$\boxed{\frac{\partial\mathcal{L}(\mathbf{w}, w_0)}{\partial w_0} = 2w_0 n - 2\sum_{i=1}^{n} y_i}$$

where $\mathbf{y}^\top\mathbf{1} \to \sum_{i=1}^{n} y_i$.

(b)

$$\mathcal{L}(\mathbf{w}, w_0) = \mathbf{y}^\top\mathbf{y} - 2\mathbf{y}^\top\mathbf{X}\mathbf{w} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} + \lambda\mathbf{w}^\top\mathbf{w} + w_0^2\mathbf{1}^\top\mathbf{1} - 2w_0(\mathbf{y}^\top\mathbf{1} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{1})$$

From this equation, which we obtained from above, we are able to cancel out a few of the terms given that they do not contain w in them. Such that the derivative becomes:

$$\frac{\partial\mathcal{L}(\mathbf{w}, w_0)}{\partial\mathbf{w}} = \underbrace{-2\frac{\partial}{\partial\mathbf{w}}\left(\mathbf{y}^\top\mathbf{X}\mathbf{w}\right)}_{\textcircled{1}} + \underbrace{\frac{\partial}{\partial\mathbf{w}}\left(\mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w}\right)}_{\textcircled{2}} + \underbrace{\frac{\partial}{\partial\mathbf{w}}\left(\lambda\mathbf{w}^\top\mathbf{w}\right)}_{\textcircled{3}} \underbrace{-2w_0\frac{\partial}{\partial\mathbf{w}}\left(\mathbf{w}^\top\mathbf{X}^\top\mathbf{1}\right)}_{\textcircled{4}}$$

Now, solving each individual term for their respective gradients w.r.t w we obtain the following:

$$\textcircled{1} \to -2\mathbf{y}^\top\mathbf{X} = -2\mathbf{X}^\top\mathbf{y}$$

$$\textcircled{2} \to 2\mathbf{X}^\top\mathbf{X}\mathbf{w}$$

$$\textcircled{3} \to \lambda 2\mathbf{w}^\top$$

$$\textcircled{4} \to -2w_0\mathbf{1}^\top\mathbf{X}^\top$$

Now by putting all of these terms back into the equation we obtain that the derivative is the following

$$\frac{\partial\mathcal{L}(\mathbf{w}, w_0)}{\partial\mathbf{w}} = \underbrace{-2\mathbf{X}^\top\mathbf{y}}_{\textcircled{1}} + \underbrace{2\mathbf{X}^\top\mathbf{X}\mathbf{w}}_{\textcircled{2}} + \underbrace{\lambda 2\mathbf{w}^\top}_{\textcircled{3}} \underbrace{-2w_0\mathbf{1}^\top\mathbf{X}^\top}_{\textcircled{4}}$$

Because the above stated assumption in that $\frac{1}{n}\sum_{i=1}^{n} x_{ij} = 0$, this implies that the last term $\textcircled{4}$ goes to zero as $\mathbf{1}^\top\mathbf{X}^\top \to 0$ because $\mathbf{1}^\top\mathbf{X}^\top$ sums up to $\sum_{i=1}^{n} x_{ij}$ for every row. This in turn gives us the final expression as

$$\frac{\partial\mathcal{L}(\mathbf{w}, w_0)}{\partial\mathbf{w}} = -2\mathbf{X}^\top\mathbf{y} + 2\mathbf{X}^\top\mathbf{X}\mathbf{w} + \lambda 2\mathbf{w}^\top$$

Which can be fully simplified using simple algebra to

$$\boxed{\frac{\partial\mathcal{L}(\mathbf{w}, w_0)}{\partial\mathbf{w}} = 2\left(-\mathbf{X}^\top\mathbf{y} + \mathbf{w}\left(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}\right)\right)}$$

(c)

In order to solve for the global optimizer of the loss function we must seek the 0's of the derivative of the loss function. From part (a) and part (b), we have obtained the derivative form with respect to finding the bias as well as finding the weight function derivatives. With that said, we must seek the optimal of both the bias and the rest of the weights independently as shown below. For the w_0 portion we begin by taking our results from part (a) and setting the derivative $\frac{\partial \mathcal{L}(\mathbf{w}, w_0)}{\partial w_0} = 0$,such that

$$\frac{\partial \mathcal{L}(\mathbf{w}, w_0)}{\partial w_0} = 2w_0 n - 2\sum_{i=1}^{n} y_i = 0$$

Which solving for w_0 gives us

$$\boxed{w_0 = \frac{1}{n}\sum_{i=1}^{n} y_i}$$

Likewise, for the w portion, we can solve for w in

$$\frac{\partial \mathcal{L}(\mathbf{w}, w_0)}{\partial \mathbf{w}} = 2\left(-\mathbf{X}^\top \mathbf{y} + \mathbf{w}\left(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}\right)\right) = 0$$

such that we can simplify this expression to

$$\mathbf{X}^\top \mathbf{y} = \mathbf{w}\left(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}\right)$$

where we move the multiplication w.r.t w on the RHS to the left by multiplying it by the inverse on both sides such that we obtain the final form as:

$$\boxed{\mathbf{w} = \left(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}\right)^{-1} \mathbf{X}^\top \mathbf{y}}$$

(d)

In order to prove the invertibility of a matrix, we must ensure that the matrix is positive definite, meaning that all eigenvalues are positive. To make sure that we can solve the above stated equation and obtain the vector of weights w for our regression, we must invert the matrix $\left(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}\right)$, which means that if this matrix is non-invertible we must seek a non-analytical approach to solving this problem. So, for this case, given that $\mathbf{X}^\top \mathbf{X}$ is semi-definite and that $\lambda > 0$, making $\lambda \mathbf{I}$ positive definite, we can infer that the total matrix, is positive definite, meaning all eigenvalues are positive eigenvalues as we are introducing the stretch though the $\lambda \mathbf{I}$ term.

In other words, we can prove this mathematically by multiplying each part by a vector such that:

$$\mathbf{X}^\top \mathbf{X} \mathbf{v} = \alpha_i \mathbf{v}$$

$$\lambda \mathbf{I} \mathbf{v} = \lambda_i \mathbf{v}$$

If we know that the eigenvalues here are $a_i \geq 0$ and that $\lambda_i > 0$, we know that $((\alpha_i + \lambda_i) > 0)$.

■

(e)

Noting that the only difference between the ridge regression and the standard last-squares regression is the regularization term ($\lambda \mathrm{w}^\top \mathrm{w}$) at the end. The difference in the last problem highlights the importance of the regularization term in the ridge regression. There are two main take-aways from this problem: 1) that the regularization increases the loss and as seen from eq. (2) decreases the value of w as it is inverse proportional. This helps make sure we are not overfitting the model to the data which would capture the noise of the data instead of the major features. 2) Also, from eq. (2) we can see that the regularization term also makes the system of equations always solvable by avoiding any singularity in the matrix - which with the regularization term becomes positive definite.

Problem 2 (Priors and Regularization,7pts)

In this problem we consider a model of Bayesian linear regression. Define the prior on the parameters as,

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \mathbf{0}, \alpha^{-1}\mathbf{I}),$$

where α is as scalar precision hyperparameter that controls the variance of the Gaussian prior. Define the likelihood as,

$$p(\mathbf{y} \mid \mathbf{w}) = \prod_{i=1}^{n} \mathcal{N}(y_i \mid \mathbf{w}^\mathsf{T}\mathbf{x}_i, \beta^{-1}),$$

where β is another fixed scalar defining the variance.

Using the fact that the posterior is the product of the prior and the likelihood (up to a normalization constant), i.e.,

$$\arg\max_{\mathbf{w}} \ln p(\mathbf{w} \mid \mathbf{y}) = \arg\max_{\mathbf{w}} \left(\ln p(\mathbf{w}) + \ln p(\mathbf{y} \mid \mathbf{w})\right).$$

Show that maximizing the log posterior is equivalent to minimizing a regularized loss function given by $\mathcal{L}(\mathbf{w}) + \lambda\mathcal{R}(\mathbf{w})$, where

$$\mathcal{L}(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \mathbf{w}^\mathsf{T}\mathbf{x}_i)^2$$

$$\mathcal{R}(\mathbf{w}) = \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w}$$

Do this by writing $\ln p(\mathbf{w} \mid \mathbf{y})$ as a function of $\mathcal{L}(\mathbf{w})$ and $\mathcal{R}(\mathbf{w})$, dropping constant terms if necessary. Conclude that maximizing this posterior is equivalent to minimizing the regularized error term given by $\mathcal{L}(\mathbf{w}) + \lambda\mathcal{R}(\mathbf{w})$ for a λ expressed in terms of the problem's constants.

Solution

We begin by defining the following equations for univariate and multivariate Gaussians, respectively:

$$\mathcal{N}(z \mid \mu, \sigma^2) = \frac{1}{\sqrt{|2\pi\sigma^2|}} \exp\left(-\frac{1}{2}(z-\mu)\sigma^{-2}(z-\mu)\right) \tag{3}$$

$$\mathcal{N}(\mathbf{z} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})\right) \tag{4}$$

Now, knowing that the prior is given by $\mathcal{N}(\mathbf{w} \mid \mathbf{0}, \alpha^{-1}\mathbf{I})$, we can obtain the multivariate distribution for these parameters using eq. (4), such that

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \mathbf{0}, \alpha^{-1}\mathbf{I}) = \frac{1}{\sqrt{|2\pi\alpha^{-1}\mathbf{I}|}} \exp\left(-\frac{1}{2}\mathbf{w}^\top\alpha\mathbf{I}(\mathbf{w})\right)$$

Taking the ln of both sides gives some nice additive properties to this equation for further inspection. Thus it becomes:

$$\ln(p(\mathbf{w})) = \ln(\mathcal{N}(\mathbf{w} \mid \mathbf{0}, \alpha^{-1}\mathbf{I})) = -\frac{1}{2}\ln\left(|2\pi\alpha^{-1}\mathbf{I}|\right) - \frac{1}{2}\mathbf{w}^\top\alpha\mathbf{I}(\mathbf{w})$$

Where we separate the constant term $-\frac{1}{2}\ln\left(|2\pi\alpha^{-1}\mathbf{I}|\right) = c_1$ away, such that

$$\ln(p(\mathbf{w})) = c_1 - \frac{1}{2}\alpha\mathbf{w}^\top\mathbf{w}$$

Likewise, for the likelihood, we take the ln of both sides and follow a similar procedure as described above using the univariate equation eq. (3) such that:

$$\begin{aligned}
\ln(p(\mathbf{y} \mid \mathbf{w})) &= \sum_{i=1}^{n} \ln(\mathcal{N}(y_i \mid \mathbf{w}^\top\mathbf{x}_i, \beta^{-1})) \\
&= \sum_{i=1}^{n} -\frac{1}{2}\left(\ln\left(|2\pi\beta^{-1}|\right) + (y_i - \mathbf{w}^\top\mathbf{x}_i)\beta(y_i - \mathbf{w}^\top\mathbf{x}_i)\right) \\
&= \sum_{i=1}^{n} -\frac{1}{2}\left(c_2 + (y_i - \mathbf{w}^\top\mathbf{x}_i)\beta(y_i - \mathbf{w}^\top\mathbf{x}_i)\right) \\
&= \sum_{i=1}^{n} -\frac{1}{2}\left(c_2 + \beta(y_i - \mathbf{w}^\top\mathbf{x}_i)^2\right)
\end{aligned}$$

Putting it all together we get the following:

$$\arg\max_{\mathbf{w}} \left(\ln p(\mathbf{w}) + \ln p(\mathbf{y} \mid \mathbf{w})\right) = \arg\max_{\mathbf{w}} \left(c_1 - \frac{1}{2}\alpha\mathbf{w}^\top\mathbf{w} - \sum_{i=1}^{n}\frac{1}{2}\left(c_2 + \beta(y_i - \mathbf{w}^\top\mathbf{x}_i)^2\right)\right)$$

Dropping the constant terms (in terms of w), as they do not contribute to the rate of change when finding the optima, we obtain:

$$\arg\max_{\mathbf{w}} \left(\ln p(\mathbf{w}) + \ln p(\mathbf{y} \mid \mathbf{w})\right) = \arg\max_{\mathbf{w}} -\left(\alpha\frac{1}{2}\mathbf{w}^\top\mathbf{w} + \beta\frac{1}{2}\sum_{i=1}^{n}(y_i - \mathbf{w}^\top\mathbf{x}_i)^2\right)$$

When comparing both the LHS and RHS of the posterior comparison, we can infer that λ in terms of α and β becomes

$$\lambda = \frac{\alpha}{\beta}$$

Such that now we have:

$$\boxed{\arg\max_{\mathbf{w}} \left(\ln p(\mathbf{w}) + \ln p(\mathbf{y} \mid \mathbf{w})\right) = \arg\max_{\mathbf{w}} -\left(\frac{1}{2}\sum_{i=1}^{n}(y_i - \mathbf{w}^\top\mathbf{x}_i)^2 + \lambda\frac{1}{2}\mathbf{w}^\top\mathbf{w}\right)} \qquad (5)$$

Comparing eq. (5) to the original equation above as a function of $\mathcal{L}(\mathbf{w})\&\mathcal{R}(\mathbf{w})$ we see that the only difference here is that there exists a negative sign. Therefore we see that maximizing the log posterior is equivalent to minimizing the regularized loss function.

■

3. Modeling Changes in Congress [10pts]

The objective of this problem is to learn about linear regression with basis functions by modeling the average age of the US Congress. The file `congress-ages.csv` contains the data you will use for this problem. It has two columns. The first one is an integer that indicates the Congress number. Currently, the 114th Congress is in session. The second is the average age of that members of that Congress. The data file looks like this:

```
congress,average_age
80,52.4959
81,52.6415
82,53.2328
83,53.1657
84,53.4142
85,54.1689
86,53.1581
87,53.5886
```

and you can see a plot of the data in Figure 1.

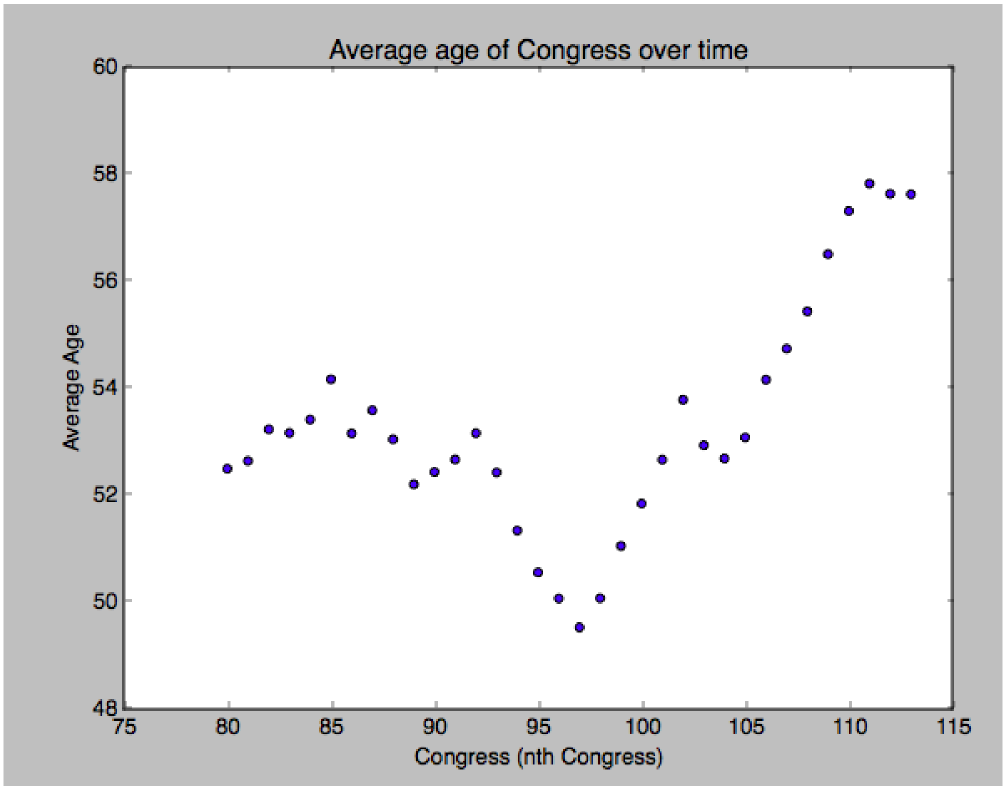

Figure 1: Average age of Congress. The horizontal axis is the Congress number, and the vertical axis is the average age of the congressmen.

Problem 3 (Modeling Changes in Congress, 10pts)

Implement basis function regression with ordinary least squares with the above data. Some sample Python code is provided in `linreg.py`, which implements linear regression. Plot the data and regression lines for the simple linear case, and for each of the following sets of basis functions:

(a) $\phi_j(x) = x^j$ for $j = 1, \ldots, 6$

(b) $\phi_j(x) = x^j$ for $j = 1, \ldots, 4$

(c) $\phi_j(x) = \sin(x/j)$ for $j = 1, \ldots, 6$

(d) $\phi_j(x) = \sin(x/j)$ for $j = 1, \ldots, 10$

(e) $\phi_j(x) = \sin(x/j)$ for $j = 1, \ldots, 22$

In addition to the plots, provide one or two sentences for each, explaining whether you think it is fitting well, overfitting or underfitting. If it does not fit well, provide a sentence explaining why. A good fit should capture the most important trends in the data.

Solution

To compare the different basis functions, we will look at the loss of function using the predetermined weights $\mathbf{w}^*$. The lower the values of the loss, the better it is fitting the data. However, we are not able to make any conclusions on overfitting the data solely based on the loss. To do that we look at the different graphs as shown below.

(a)

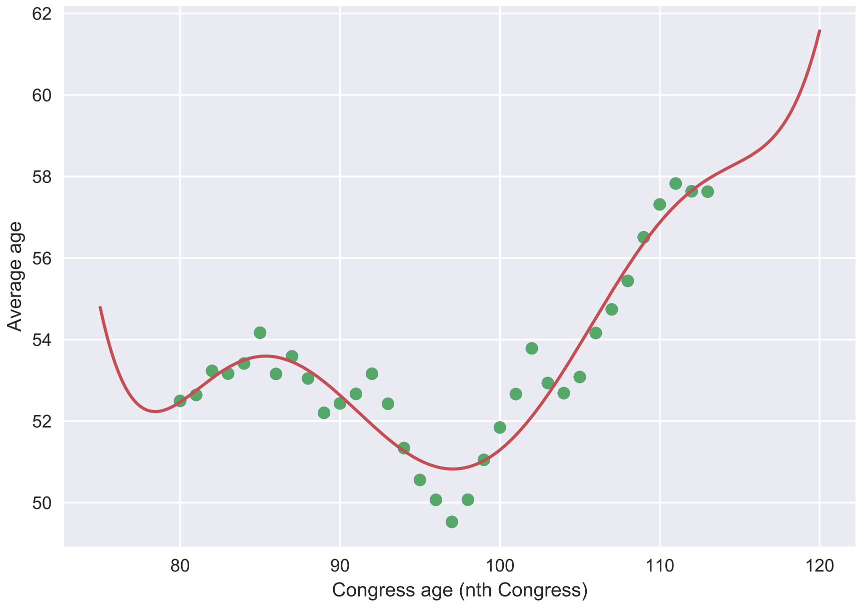

Figure 2

$$\mathcal{L}(\mathbf{w}^*)_{\text{a}} \approx 6.56$$

From this fit, we can see that the loss is not very low compared to the other fits. We also notice that some major features are not accounted for by this fit.

(b)

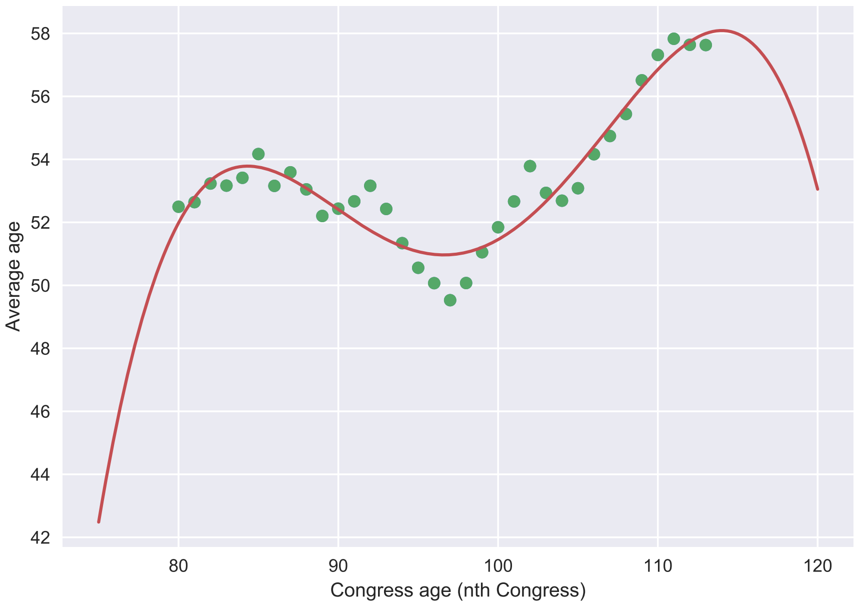

Figure 3

$$\mathcal{L}(\mathbf{w}^*)_{\mathrm{b}} \approx 7.01$$

Based on the loss, we see that this regression fits the data the least of all other basis functions. We can see that it misses a lot of the small features of the data and over-simplifies the model, not capturing enough complexities. This basis function only captures two maxima and one minimum within the region of the data.

(c)

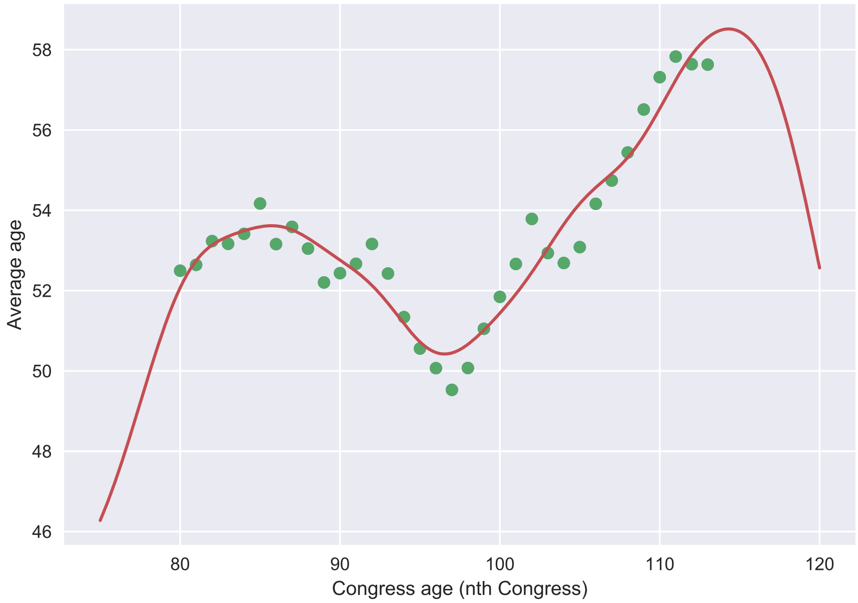

Figure 4

$$\mathcal{L}(\mathbf{w}^*)_c \approx 5.71$$

This basis function does a good job not overfitting the data compared to the other sinusoidal basis functions. Even though this function has a higher value of loss, it seem to be doing a good job not overfitting the data.

(d)

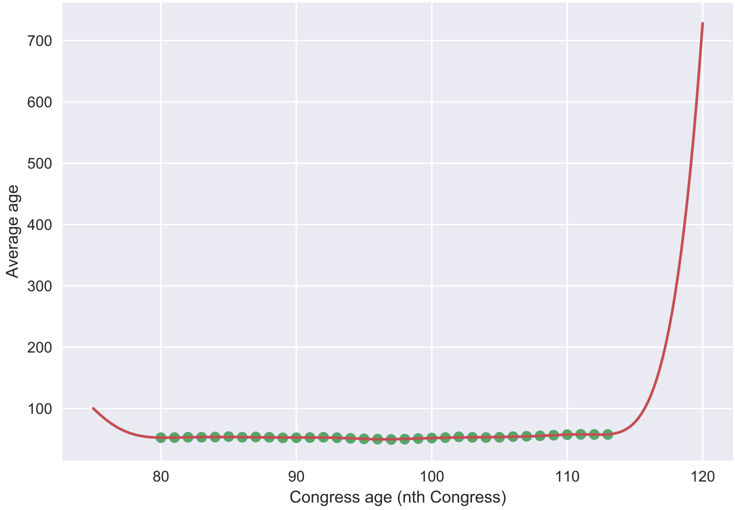

Figure 5

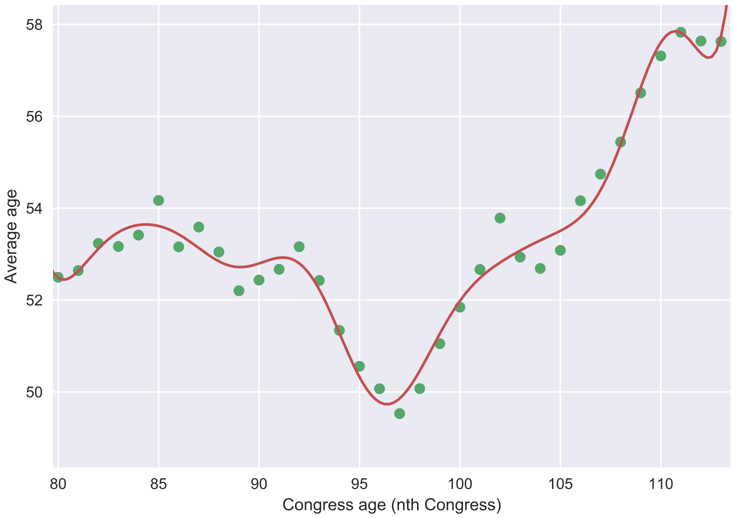

Figure 6

$$\mathcal{L}(\mathbf{w}^*)_{\mathrm{d}} \approx 1.93$$

In fig. 5 we can see that the regression function does not do a very good job modeling the average age as a prediction for future congresses. However, from fig. 6, we can see that the model does a fairly good job capturing the age distribution for the data we have for the congress 80 to 115. I would trust this model to obtain data for the current congresses as a model, but would not trust it to make future predictions as to what the average age of congress will look like in the future.

(e)

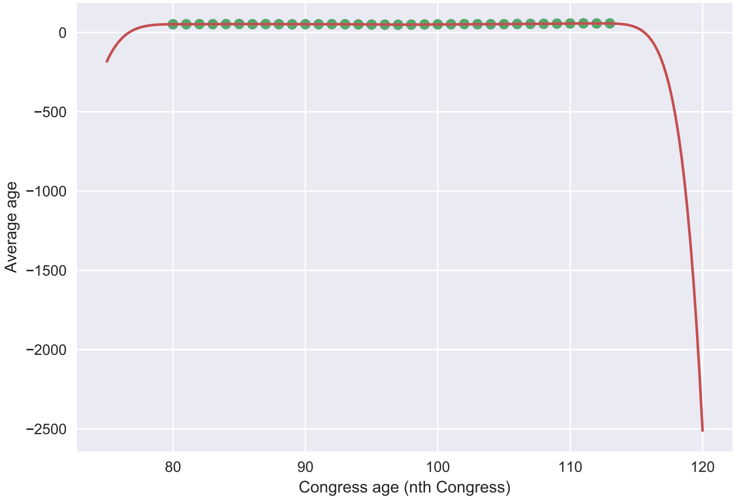

Figure 7

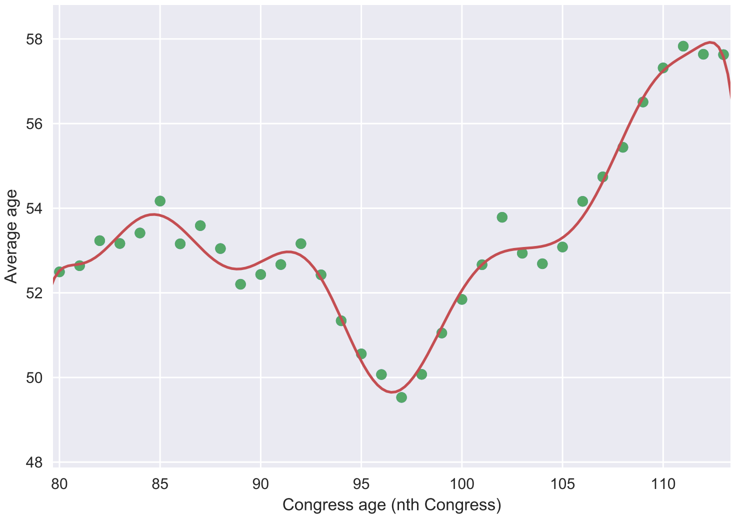

Figure 8

$$\mathcal{L}(\mathbf{w}^*)_\mathrm{e} \approx 1.45$$

In this case we see a similar result as discussed in part (d), in that this model does a very good job capturing the features of the current data whereas it does not do a good job predicting the average age of the congresses in the future. Another feature to notice is that even though we added complexity to the basis functions, we did not gain that complexity back in form of accuracy. The plot does not show any significant benefit between what we were able to achieve in part (d) versus here. We can see that by just adding more terms to our basis functions, we cannot increase the accuracy of our model, but instead, if we want to make our models better we must choose different basis functions or a different model other than linear regression all together.

Code for Problem 3

```
#####################
# CS 181, Spring 2016
# Homework 1, Problem 3
#
##################

import csv
import seaborn
import numpy as np
import matplotlib.pyplot as plt

# Creating the different basis functions according to the PSET
def p3a(times): return np.ones(times.shape),times,times**2.,times**3.,times**4.,times**5.,times**6.
def p3b(times): return np.ones(times.shape),times,times**2.,times**3.,times**4.
def p3c(times): return np.ones(times.shape),np.sin(times/1.),np.sin(times/2.),np.sin(times/3.),np.sin(times/4.),np.sin(times/5.),np.sin(times/6.)
def p3d(times): return np.ones(times.shape),np.sin(times/1.),np.sin(times/2.),np.sin(times/3.),np.sin(times/4.),np.sin(times/5.),np.sin(times/6.),np.sin(times/7.),np.sin(times/8.),np.sin(times/9.),np.sin(times/10.)
def p3e(times): return np.ones(times.shape),np.sin(times/1.),np.sin(times/2.),np.sin(times/3.),np.sin(times/4.),np.sin(times/5.),np.sin(times/6.),np.sin(times/7.),np.sin(times/8.),np.sin(times/9.),np.sin(times/10.),np.sin(times/11.),np.sin(times/12.),np.sin(times/13.),np.sin(times/14.),np.sin(times/15.),np.sin(times/16.),np.sin(times/17.),np.sin(times/18.),np.sin(times/19.),np.sin(times/20.),np.sin(times/21.),np.sin(times/22.)


# Change for a particular part of the problem
def basis(times): return p3e(times)

csv_filename = 'congress-ages.csv'
times  = []
ages = []

with open(csv_filename, 'r') as csv_fh:

    # Parse as a CSV file.
    reader = csv.reader(csv_fh)

    # Skip the header line.
    next(reader, None)

    # Loop over the file.
    for row in reader:

        # Store the data.
        times.append(float(row[0]))
        ages.append(float(row[1]))

# Turn the data into numpy arrays.
times  = np.array(times)
ages = np.array(ages)

# Plot the data.
plt.plot(times, ages, 'o')
plt.xlabel("Congress age (nth Congress)")
plt.ylabel("Average age")
# plt.show()

# Create the simplest basis, with just the time and an offset.

X = np.vstack((basis(times))).T
```

```python

# Nothing fancy for outputs.
Y = ages

# Find the regression weights using the Moore-Penrose pseudoinverse.
w = np.linalg.solve(np.dot(X.T, X) , np.dot(X.T, Y))

# Compute the regression line on a grid of inputs.
# DO NOT CHANGE grid_times!!!!!
grid_times = np.linspace(75, 120, 200)
grid_X = np.vstack(basis(grid_times))
grid_Yhat  = np.dot(grid_X.T, w)

# Plot the data and the regression line.
plt.plot(times, ages, 'o', grid_times, grid_Yhat, '-')
plt.xlabel("Congress age (nth Congress)")
plt.ylabel("Average age")
# plt.show()
# plt.savefig('images/p3e.pdf', bbox_inches='tight')

# Computing the Loss for each part to see how well the model is fitting the data
Loss=0
for i in xrange(len(X)):
  Loss+=1./2.*(ages[i]-np.dot(basis(times[i]),w))**2
print 'The loss using this basis function is',Loss
```

Listing 1: Python Code for Problem 3

Problem 4 (Calibration, 1pt)

Approximately how long did this homework take you to complete?

Answer: Approximately 7 hours.