COMS W4721: Machine Learning for Data Science

Homework - 1.

Problem - 1

(a) Joint likelihood of $(x_1, x_2, \dots x_N)$

$$p(x_1, x_2, \dots x_N / \pi, r) = \prod_{i=1}^{N} \binom{x_i + r - 1}{x_i} \pi^{x_i} (1-\pi)^r$$

$(\because x_i$s are iids$)$

$$= \pi^{\sum_{i=1}^{N} x_i} (1-\pi)^{nr} \prod_{i=1}^{N} \binom{x_i + r - 1}{x_i}$$

(b) $$\text{MLE } (\hat{\pi}) = \underset{\pi}{\text{Arg max}}\ p(x_1, x_2, \dots x_N / \pi, r)$$

$$= \underset{\pi}{\text{Arg max}}\ \ln p(x_1, x_2, \dots x_N / \pi, r)$$

$$\therefore \nabla \ln p(x_1, x_2, \dots x_N / \pi, r) = 0$$

$$\Rightarrow \nabla \left(\sum_{i=1}^{N} x_i\right) \ln \pi + nr \ln(1-\pi) + \sum_{i=1}^{N} \ln \binom{x_i + r - 1}{x_i} = 0$$

$$\Rightarrow \frac{\left(\sum_{i=1}^{N} x_i\right)}{\pi} + \frac{nr}{(1-\pi)}(-1) + 0 = 0$$

$$\frac{nr}{1-\pi} = \frac{\sum_{i=1}^{N} x_i}{\pi}$$

$$nr\pi = \left(\sum_{i=1}^{N} x_i\right) - \pi \left(\sum_{i=1}^{N} x_i\right)$$

$$\pi\left(nr + \left(\sum_{i=1}^{N} x_i\right)\right) = \sum_{i=1}^{N} x_i$$

$$\boxed{\hat{\pi}_{MLE} = \frac{\sum_{i=1}^{N} x_i}{nr + \sum_{i=1}^{N} x_i}}$$

(c) prior dist $p(\pi) = \text{Beta}(a, b)$

posterior: $p(\pi / x_1, x_2, \ldots x_N) = \dfrac{p(x_1, \ldots x_N / \pi)\, p(\pi)}{\int_0^1 p(x_1, x_2 \ldots x_N / \pi)\, p(\pi)\, d\pi}$

$$\pi_{MAP} = \text{Arg}\max_{\pi}\; p(\pi / x_1, x_2, \ldots x_N)$$

$$= \text{Arg}\max_{\pi}\; \ln p(\pi / x_1, \ldots x_N)$$

$$\Rightarrow \nabla_{\pi} \ln p(\pi / x_1, \ldots x_N) = 0$$

$$\Rightarrow \nabla_{\pi}\; \ln p(x_1, \ldots x_N / \pi) + \ln p(\pi) - \ln k = 0$$

$$\Rightarrow \nabla_{\pi} \left(\sum_{i=1}^{N} x_i\right) \ln \pi + nr \ln(1-\pi) + \sum_{i=1}^{N} \ln \binom{x_i + r - 1}{x_i} + \ln \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} + (a-1)\ln \pi + (b-1)\ln(1-\pi) = 0$$

$$\Rightarrow \frac{\sum_{i=1}^{N} x_i}{\pi} + \frac{nr}{1-\pi}(-1) + \frac{a-1}{\pi} + \frac{b-1}{1-\pi}(-1) = 0$$

$$\Rightarrow \quad \frac{\sum_{i=1}^{N} x_i + a - 1}{\pi} = \frac{nr + b - 1}{1-\pi}$$

$$\Rightarrow \quad \hat{\pi}_{MAP} = \frac{\left(\sum_{i=1}^{N} x_i\right) + a - 1}{\left(\sum_{i=1}^{N} x_i\right) + a + b + nr - 2}$$

(d) $$p(\pi / x_1, x_2, \dots x_N) = \frac{p(x_1, \dots x_N / \pi)\, p(\pi)}{\int_0^1 p(x_1, \dots x_N/\pi)\, p(\pi)\, d\pi}$$

$$\int_0^1 p(x_1, \dots x_N/\pi)\, p(\pi)\, d\pi = \left[\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \prod_{i=1}^{N} \binom{x_i + r - 1}{x_i}\right] \int_0^1 \pi^{\Sigma x_i} (1-\pi)^{nr}\, \pi^{a-1} (1-\pi)^{b-1}\, d\pi$$

$$= \left[\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \prod_{i=1}^{N} \binom{x_i + r - 1}{x_i}\right] \int_0^1 \pi^{\Sigma x_i + a - 1} (1-\pi)^{nr + b - 1}\, d\pi$$

$$= \left[\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \prod_{i=1}^{N} \binom{x_i + r - 1}{x_i}\right] \frac{\Gamma((\Sigma x_i) + a)\,\Gamma(nr + b)}{\Gamma(\Sigma x_i + a + nr + b)}$$

$$p(\pi/x_1, \dots x_N) = \frac{\cancel{\prod_{i=1}^{N} \binom{x_i + r - 1}{x_i}} \cdot \pi^{\Sigma x_i} (1-\pi)^{nr}\, \cancel{\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}} \cdot \pi^{a-1} (1-\pi)^{b-1}}{\left[\cancel{\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}}\, \cancel{\prod_{i=1}^{N} \binom{x_i + r - 1}{x_i}}\right] \frac{\Gamma(\Sigma x_i + a)\,\Gamma(nr + b)}{\Gamma(\Sigma x_i + a + nr + b)}}$$

$$p(\pi/x_1,\dots x_N) = \frac{\Gamma\left(\sum x_i + a + nr + b\right)}{\Gamma\left(\sum x_i + a\right)\Gamma(nr+b)} \cdot \pi^{\sum x_i + a - 1}(1-\pi)^{nr+b-1}$$

$$\sim \text{Beta}\left(\sum x_i + a,\ nr + b\right)$$

∴ posterior is beta distributed.

ⓔ $$E(\pi/x_1,\dots x_N) = \frac{\Gamma\left(\sum x_i + a + nr + b\right)}{\Gamma\left(\sum x_i + a\right)\Gamma(nr+b)} \int_0^1 \pi^{\sum x_i + a - 1}(1-\pi)^{nr+b-1}\,d\pi\ .\pi$$

$$= \frac{\Gamma\left(\sum x_i + a + nr + b\right)}{\Gamma\left(\sum x_i + a\right)\Gamma(nr+b)} \cdot \int_0^1 \pi^{(\sum x_i + a + 1) - 1}(1-\pi)^{nr+b-1}\,d\pi$$

$$= \frac{\Gamma\left(\sum x_i + a + nr + b\right)}{\Gamma\left(\sum x_i + a\right)\cancel{\Gamma(nr+b)}} \cdot \frac{\Gamma\left(\sum x_i + a + 1\right)\cancel{\Gamma(nr+b)}}{\Gamma\left(\sum x_i + a + nr + b + 1\right)}$$

$$= \frac{\cancel{\Gamma\left(\sum x_i + a + nr + b\right)} \cdot \left(\sum x_i + a\right) \cdot \cancel{\Gamma\left(\sum x_i + a\right)}}{\cancel{\Gamma\left(\sum x_i + a\right)} + \left(\sum x_i + a + nr + b\right)\cancel{\Gamma\left(\sum x_i + a + nr + b\right)}}$$

$$= \frac{\left(\sum_{i=1}^{N} x_i\right) + a}{\sum_{i=1}^{N} x_i + a + nr + b}$$

$$E\left(\pi^2/x_1,\dots x_N\right) = \int_0^1 \pi^2\, p\left(\pi/x_1,\dots x_N\right) d\pi$$

$$= \frac{\Gamma\left(\sum x_i + a + nr + b\right)}{\Gamma\left(\sum x_i + a\right)\Gamma(nr+b)} \int_0^1 \pi^{\sum x_i + a + 1} (1-\pi)^{nr+b-1}\, d\pi$$

$$= \frac{\Gamma\left(\sum x_i + a + nr + b\right)}{\Gamma\left(\sum x_i + a\right)\Gamma(nr+b)} \cdot \frac{\Gamma\left(\sum x_i + a + 2\right)\Gamma(nr+b)}{\Gamma\left(\sum x_i + a + nr + b + 2\right)}$$

$$= \frac{\Gamma\left(\sum x_i + \cancel{a + nr} + b\right)}{\Gamma\left(\cancel{\sum x_i + a}\right)\Gamma\left(\cancel{nr+b}\right)} \cdot \frac{\left(\sum x_i + a + 1\right)\overset{(\sum x_i + a)}{\cancel{\left(\sum x_i + a\right)}}\,\Gamma\left(\cancel{\sum x_i + a}\right)\Gamma\left(\cancel{nr+b}\right)}{\left(\sum x_i + a + nr + b + 1\right)\left(\sum x_i + a + nr + b\right)\Gamma\left(\sum x_i + \cancel{a + nr} + b\right)}$$

$$= \frac{\left(\sum x_i + a + 1\right)\left(\sum x_i + a\right)}{\left(\sum x_i + a + nr + b + 1\right)\left(\sum x_i + a + nr + b\right)}$$

** let $\left(\sum x_i + a\right) = \alpha$ and $(nr+b) = \beta$ **

$$\mathrm{var}\left(\pi/x_1,\dots x_N\right) = E(\pi^2) - \left(E(\pi)\right)^2$$

$$= \frac{\alpha(\alpha+1)}{(\alpha+\beta+1)(\alpha+\beta)} - \frac{\alpha^2}{(\alpha+\beta)^2}$$

$$= \frac{\alpha(\alpha+\beta)(\alpha+1) - \alpha^2(\alpha+\beta+1)}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

$$= \frac{\cancel{\alpha^3} + \cancel{\alpha^2} + \cancel{\alpha^2\beta} + \alpha\beta - \cancel{\alpha^3} - \cancel{\alpha^2\beta} - \cancel{\alpha^2}}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

$$\boxed{Var(\pi/x_1..x_N) = \frac{(\sum x_i + a)(n\gamma + b)}{(\sum x_i + a + n\gamma + b)^2(\sum x_i + a + n\gamma + b + 1)}}$$

<u>Relationship</u>:

$$\hat{\pi}_{MAP} = \frac{\left(\sum_{i=1}^{N} x_i\right) + a - 1}{\left(\sum_{i=1}^{N} x_i\right) + a + b + n\gamma - 2} \qquad E(\pi/x_1, x_2, .\, x_N) = \frac{\sum x_i + a}{\sum x_i + a + n\gamma + b}$$

for high values of N (High number of ~~samples~~ successes)

$$\left(\sum_{i=1}^{N} x_i\right) + a - 1 \simeq \sum_{i=1}^{N} x_i + a \quad \&$$

$$\left(\sum x_i\right) + a + b + n\gamma - 2 \simeq \sum_{i=1}^{N} x_i + a + n\gamma + b$$

$$\therefore \quad \underline{\hat{\pi}_{MAP} \simeq E[\pi/x_1, \ldots x_N]}.$$

$$\hat{\pi}_{MLE} = \frac{\sum_{i=1}^{N} x_i}{\sum_{i=1}^{N} x_i + n\gamma}$$

for $a = b = 0$. ie $p(\pi) = Beta(0,0)$

then $\underline{\hat{\pi}_{MLE} = E(\pi/x_1, x_2 \ldots x_N)}$

COMS 4721: Machine Learning for Data Science

Vinayak Bakshi, Columbia University, Spring 2017

February 5, 2017

Problem: 2

Part: A (a)

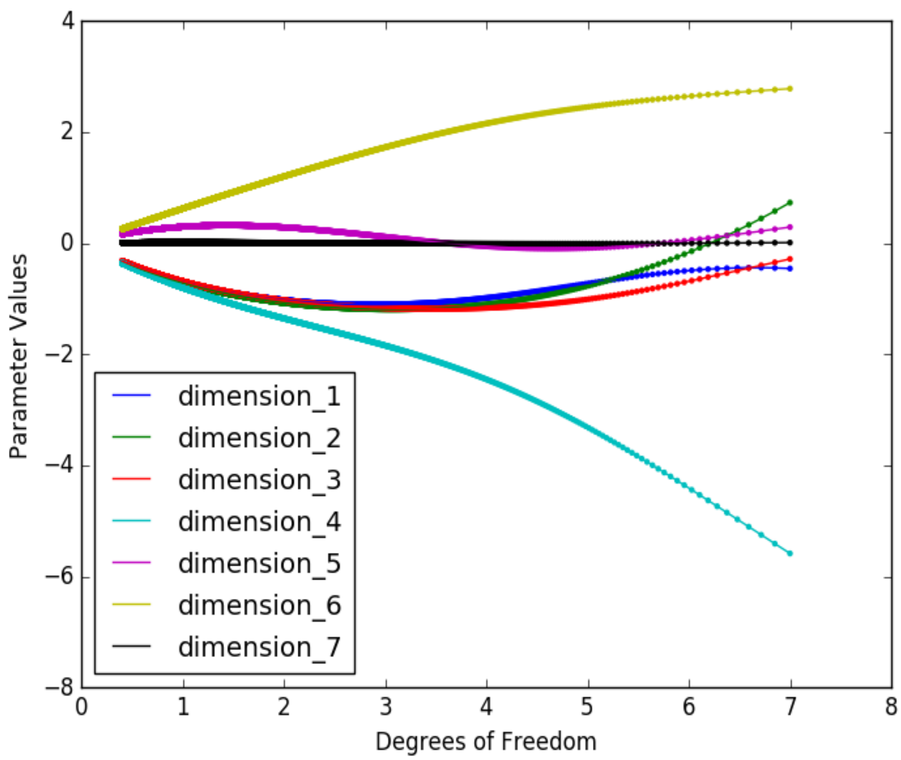

Figure 1: Variation in parameters (w_i) with Degrees of freedom $d(f)$

Part: A (b)

We can clearly see that the 4th (car weight) and 6th dimension (car yearly) stand out. This shows that the 4th dimension and 6th dimension influence the dependent variable predominantly. These dimensions are penalized the most when the regularization parameter increases.

Part: A (c)

The RMSE for $\lambda = 0$ is lowest and therefore we should pick least squares over ridge regression.

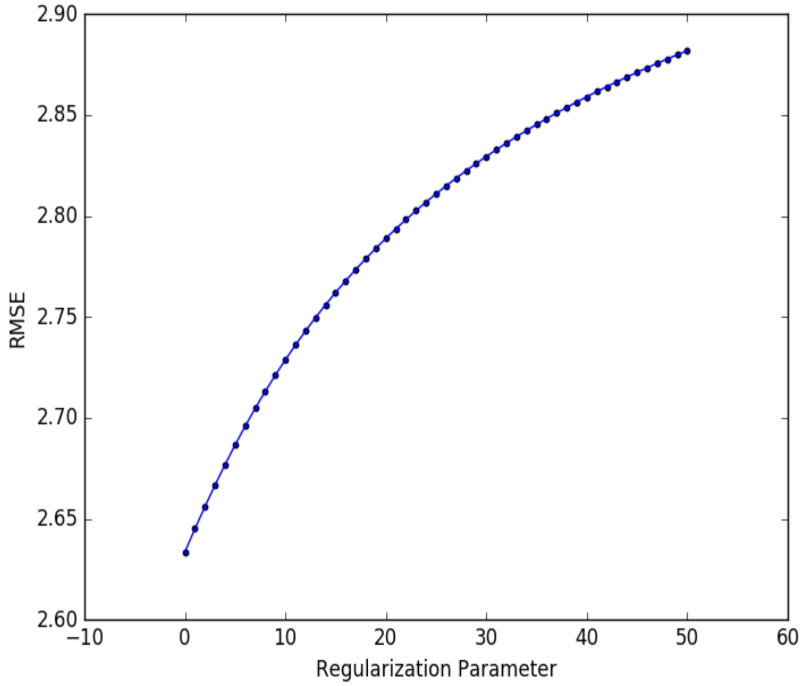

Figure 2: Variation in RMSE (w_i) with regularization paramter $d(f)$

Part: A (d)

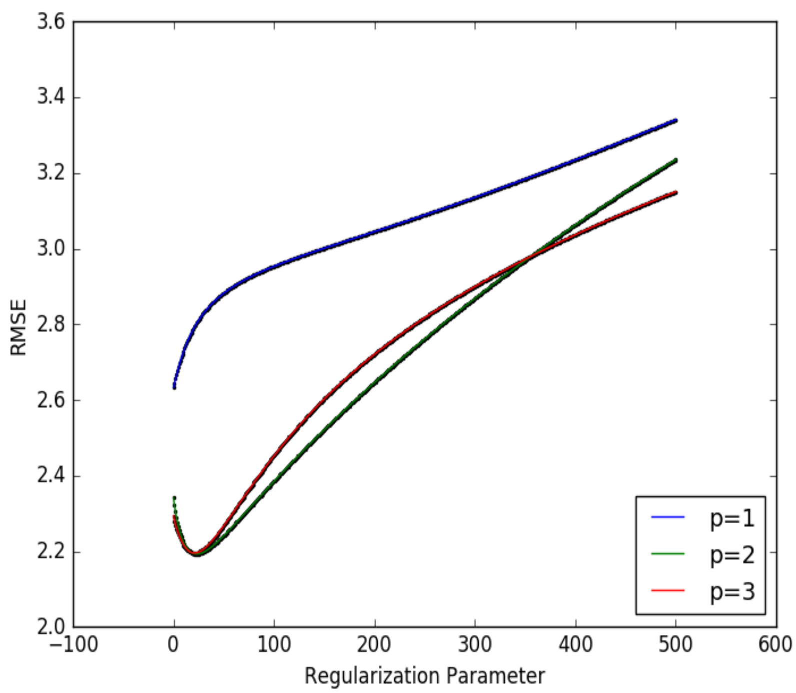

Figure 3: Variation in RMSE with regularization parameter for different polynomial degrees

We should choose the **polynomial of degree 2**. This is because it has a minimum RMSE of 2.192 for a **regularization parameter = 21**.

Python Code:

```
# import libraries
import numpy as np
import numpy.linalg as nplg
import matplotlib.pyplot as plt

#Import Data as Numpy Matrices
X_train = np.genfromtxt ('X_train.csv', delimiter=",")
Y_train = np.genfromtxt ('y_train.csv',delimiter=",")
X_test = np.genfromtxt ('X_test.csv', delimiter=",")
Y_test = np.genfromtxt ('y_test.csv',delimiter=",")

#Obtain df and weights vector
I=np.identity(len(X_train[0]))

#Intializing variables

Lambda=0
W_rr_final=[]
df=[]

# Single Value Decomposition of X_train Matrix

U, s, V = np.linalg.svd(X_train, full_matrices=True)

for Lambda in xrange(5001):

    W_rr=np.dot(np.dot((np.linalg.inv(np.dot(Lambda,I) + np.dot((
        X_train.transpose()),X_train))),(X_train.transpose())),Y_train)
    W_rr_final.append(W_rr)
    M=np.linalg.inv((Lambda*(np.linalg.inv(np.dot(np.diag(s),np.diag(s)
        ))) + I))
    df.append(np.trace(M))

W_rr_final=np.array(W_rr_final)

#plot degrees of freedom with parameter values
colors = [i for i in 'bgrcmyk']
label = ['dimension '+str(i) for i in xrange(1,8)]

for i in xrange(7):
    plt.scatter(np.array(df),W_rr_final[:,i],c=colors[i],edgecolor='',s
        =10)
    plt.plot(np.array(df),W_rr_final[:,i])

plt.legend(label,loc=3)
plt.xlabel("Degrees of Freedom")
plt.ylabel("Parameter Values")
plt.show()


#Testing for Lambda 0,1,..50
```

```
I=np.identity(7)
RMSE=[]

for Lambda in xrange(51):
    W_rr=np.dot(np.dot((np.linalg.inv(np.dot(Lambda,I) + np.dot((
        X_train.transpose()),X_train))), (X_train.transpose())),Y_train)
    Y_pred=np.dot(X_test,W_rr)
    RMSE.append(np.sqrt((1.0/42)*sum([i*i for i in Y_test-Y_pred])))

plt.scatter(xrange(51),RMSE,s=10)
plt.plot(xrange(51),RMSE)
plt.xlabel("Regularization Parameter")
plt.ylabel("RMSE")
# plt.title("Variation of RMSE with Regularization Parameter")
plt.show()


#Polynoial p=1,2,3
label=['p=1','p=2','p=3']

X_train_sqr=np.power(X_train[:,0:6],2)
X_train_cube= np.power(X_train[:,0:6],3)

X_test_sqr=np.power(X_test[:,0:6],2)
X_test_cube= np.power(X_test[:,0:6],3)

X_train_1=X_train
X_train_2=np.concatenate((X_train_1,X_train_sqr),axis=1)
X_train_3=np.concatenate((X_train_2,X_train_cube),axis=1)

X_test_1=X_test
X_test_2=np.concatenate((X_test_1,X_test_sqr),axis=1)
X_test_3=np.concatenate((X_test_2,X_test_cube),axis=1)

for Xtrain,Xtest in [(X_train_1,X_test_1),(X_train_2,X_test_2),(
    X_train_3,X_test_3)]:
    RMSE=[]
    I=np.identity(len(Xtrain[0]))
    for Lambda in xrange(501):
        W_rr=np.dot(np.dot((np.linalg.inv(np.dot(Lambda,I) + np.dot((
            Xtrain.transpose()),Xtrain))),(Xtrain.transpose())),Y_train)
        Y_pred=np.dot(Xtest,W_rr)
        RMSE.append(np.sqrt((1.0/42)*sum([i*i for i in Y_test-Y_pred]))
            )


    plt.scatter(xrange(501),RMSE,s=2)
    plt.plot(xrange(501),RMSE)
plt.xlabel("Regularization Parameter")
plt.ylabel("RMSE")
# plt.title("RMSE Vs. Regularization Parameter for p th order
    polynomial regression")
plt.legend(label,loc=4)
plt.show()
```