1003 HW4

Long Chen
lc3424@nyu.edu

April 1st, 2021

Q1

Easily observe that:

$$\hat{w} = \arg\max_{w} \sum_{i=1}^{n} log(1 + exp(-y_i w^T x_i)).$$

Also:

$$-logP = -\sum_{i=1}^{n} y_i log(1+exp(-y_i w^T x_i))+(1-y_i) \left[log(1 + exp(-y_i w^T x_i)) - log(exp(-y_i w^T x_i))\right].$$

Let,

$$f_1(x_i, y_i) = log(1 + exp(-y_i w^T x_i)),$$

$$f_2(x_i, y_i) = y_i log(1+exp(-y_i w^T x_i))+(1-y_i) \left[log(1 + exp(-y_i w^T x_i)) - log(exp(-y_i w^T x_i))\right].$$

We will show that $f_1(x_i, y_i)$ and $f_2(x_i, y_i)$ are equivalent with Bernoulli setting. Now condiser f_1: if $y_i = 1$, $f_1 = log(1 + exp(-w^T x_i))$; if $_i = -1$, $f_1 = log(1 + exp(w^T x_i))$

Now consider f_2: if $y_i = 1$, $f_2 = log(1 + exp(-w^T x_i))$; if $_i = -1$,

$$\begin{aligned} f_2 &= log(1 + exp(-w^T x_i)) - log(exp(-w^T x_i)) \\ &= log\left(\frac{1 + exp(-w^T x_i)}{exp(-w^T x_i)}\right) \\ &= log(1 + exp(w^T x_i)) \end{aligned}$$

Thus f_1, f_2 are equivalent and we say that maximizing likelihood is equivalent to minimize negative log likelihood.

Q2

Decision boundary is such that the two classes have equal probability on the boundary. Consider:

$$P(y=1|x,w) = \frac{1}{2}$$

$$\frac{1}{1+exp(-w^Tx)} = \frac{1}{2}$$

Thus $w^Tx = 0$, is the decision boundary.

Q3

$$\begin{aligned}\mathcal{L}(cw) &= log(1+exp(-cx^Tw)) + cx^Tw - cyx^Tw \\ &= log(1+exp(cx^Tw)) - cyx^Tw\end{aligned}$$

$$\begin{aligned}\frac{\delta\mathcal{L}(cw)}{\delta c} &= \frac{x^Twexp(cx^Tw)}{1+exp(cx^Tw)} - yx^Tw \\ &= x^Tw\left[\frac{1}{1+exp(cx^Tw)} - y\right]\end{aligned}$$

Since we are at decision boundary such taht $x^Tw = 0$, $\frac{\delta\mathcal{L}(cw)}{\delta c} = 0$. Therefore, $\mathcal{L}$ is invariant with respect to c.

Q4

By Rosenberg 3.1.3, $(0, -y^{(i)}w^Tx^{(i)}) \rightarrow log(exp(0) + exp(-y^{(i)}w^Tx^{(i)}))$ is convex. Also since that the norm function is convex and that the sum of convex function is also convex, we conclude that the objective function is convex.

Q5

```python
def f_objective(theta, X, y, l2_param=1):
    res = 0

    for i in range(X.shape[0]):
        res += np.logaddexp(0, -np.asarray([y[i]]).reshape(-1,1) @
    theta.reshape(1,-1) @ X[i, :])

    res /= X.shape[0]
    res += l2_param * np.power(np.linalg.norm(theta), 2)

    return res[0]
```

Q6

```python
def fit_logistic_reg(X, y, objective_function, l2_param=1):
    p = partial(f_objective, X=X, y=y, l2_param=l2_param)
    np.random.seed(42)
    init = np.random.randn(X.shape[1])
    res = minimize(p, init)

    return res.x
```

Q7

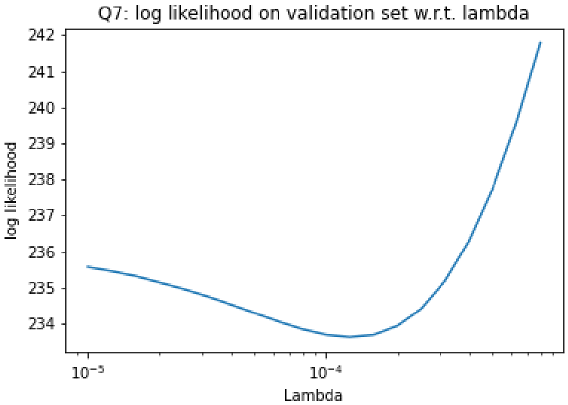
```python
x_train, y_train = np.loadtxt("X_train.txt", delimiter=’,’), np.loadtxt("y_train.txt", delimiter=’,’)
x_test, y_test = np.loadtxt("X_val.txt", delimiter=’,’), np.loadtxt("y_val.txt", delimiter=’,’)
y_train = np.where(y_train==0, -1, y_train)
y_test = np.where(y_test==0, -1, y_test)
x_train = normalize(x_train, axis=1, norm=’l1’)
x_test = normalize(x_test, axis=1, norm=’l1’)
x_train = np.hstack((np.ones((x_train.shape[0], 1)), x_train))
x_test = np.hstack((np.ones((x_test.shape[0], 1)), x_test))

def log_likelihood(theta, X, y):
    res = 0
    for i in range(X.shape[0]):
        res += np.logaddexp(0, -np.asarray([y[i]]).reshape(-1,1) @ theta.reshape(1,-1) @ X[i, :])
    return res[0]

params = [1e-9, 1e-8, 1e-7, 1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 0.1, 1, 10, 100, 1000, 10000]
res = []
for param in params:
    theta = fit_logistic_reg(x_train, y_train, f_objective, l2_param=param)
    ll = log_likelihood(theta, x_test, y_test)
    print("val log likelihood for lambda = {} is {}".format(param, ll))
    res.append(ll)

params = np.power(10, np.arange(-5,-3,0.1))
res = []
for param in params:
    theta = fit_logistic_reg(x_train, y_train, f_objective, l2_param=param)
    ll = log_likelihood(theta, x_test, y_test)
    print("val neg log likelihood for lambda = {} is {}".format(param, ll))
    res.append(ll)

plt.xscale(’log’)
plt.plot(params, res)
plt.ylabel(’log likelihood’)
plt.xlabel('Lambda')
plt.title('Q7: log likelihood on validation set w.r.t. lambda')
plt.savefig('Q7')
plt.show()
```



Best regularization term = 0.00012589254117941558 , with negative log likelihood = 233.62087156076953.

Q9

Since $p(w|\mathcal{D}) = \frac{p(\mathcal{D})p(w)}{p(\mathcal{D})}$, we can conclude that:

$$p(w|\mathcal{D}) \propto e^{-NLL_{\mathcal{D}}(w)}p(w)$$

Q10

No. From Q9, the likelihood of logistic regression is in exponential family, where the conjugate prior should be in gamma distribution instead of Gaussian distribution. Also, in the absence of summation in likelihood, $e^{-NLL_{\mathcal{D}}(w)}p(w)$ will not get a posterior in exponential family.

Q11

$$log(p(w|\mathcal{D})) = -log(\sqrt{2\pi\Sigma}) - \sum_{i=1}^{n} log(1 + exp(-y_i w^T x_i)) - \frac{1}{2}\frac{w^T w}{\Sigma}$$

$$\begin{aligned} w_{MAP} &= \arg\max log(p(w|\mathcal{D})) \\ &= \arg\min \sum_{i=1}^{n} log(1 + exp(-y_i w^T x_i)) + \frac{1}{2}\frac{w^T w}{\Sigma} \end{aligned}$$

We also have,

$$\begin{aligned} w_{MLE} &= \arg\min J(w) \\ &= \arg\min \sum_{i=1}^{n} log(1 + exp(-y_i w^T x_i)) + \lambda \|w\|^2 \end{aligned}$$

Set the two terms equal, we get:

$$\frac{1}{2\sigma} = \lambda n I$$

$$\Sigma = (2\lambda n I)^{-1}$$

where n is the number of datapoints.

Q12

Following Q11,

$$I = (2\lambda n I)^{-1}$$

$$\lambda = \frac{1}{2n}$$

where n is the number of datapoints.

Q13

$$\begin{aligned} p(x = H|\theta_1, \theta_2) &= p(x = H|\theta_1)p(x = H|z = H, \theta_2) + p(x = T|\theta_1)p(x = H|z = T) \\ &= p(x = H|\theta_1)p(x = H|z = H, \theta_2) + p(x = T|\theta_1) \times 0 \\ &= \theta_1\theta_2 \end{aligned}$$

Q14

$$\begin{aligned} p(x = T|\theta_1, \theta_2) &= p(x = T|\theta_1)p(x = T|z = T) + p(z = H|\theta_1)p(x = T|z = H, \theta_2) \\ &= (1 - \theta_1) \times 1 + \theta_1(1 - \theta_2) \\ &= 1 - \theta_1\theta_2 \end{aligned}$$

Thus:

$$\begin{aligned} p(\mathcal{D}_r) &= [p(x = H|\theta_1, \theta_2)]^{n_h} [p(x = T|\theta_1, \theta_2)]^{n_t} \\ &= (\theta_1\theta_2)^{n_h}(1 - \theta_1\theta_2)^{n_t} \end{aligned} \tag{1}$$

Q15

No. From (1) we see that we can find optimal $\theta_1\theta_2$, but we will not be able to find individual, optimal θ's.

Q16

Since $\mathcal{D}_r$ and $\mathcal{D}_c$ are independent,

$$\begin{aligned}\mathcal{L}(\theta_1,\theta_2) &= p(\mathcal{D}_r,\mathcal{D}_c|\theta_1,\theta_2)\\ &= p(\mathcal{D}_r|\theta_1,\theta_2)p(\mathcal{D}_c|\theta_1,\theta_2)\\ &= (\theta_1\theta_2)^{n_h}(1-\theta_1\theta_2)^{n_t}\theta_1^{c_h}(1-\theta_1)^{c_t}\end{aligned}$$

$$log\mathcal{L}(\theta_1,\theta_2) = n_h log(\theta_1) + n_h log(\theta_2) + n_t log(1-\theta_1\theta_2) + c_h log(\theta_1) + c_t log((1-\theta_1))$$

By taking derivatives of LL w.r.t θ_1 and θ_2, we have:

$$\hat{\theta}_1 = \frac{c_h}{c_h+c_t}$$

$$\hat{\theta}_2 = \frac{\frac{n_h}{n_h+n_t}}{\hat{\theta}_1} = \frac{n_hc_h+n_hc_t}{c_hn_h+n_tc_h}$$

Q17

$$\begin{aligned}p(\theta_1|\mathcal{D}) &\propto p(\mathcal{D}_c|\theta_1)p(\theta_1)\\ &= \theta_1^{c_h}(1-\theta_1)^{c_t}\theta_1^{h-1}(1-\theta_1)^{t-1}\\ &= \theta_1^{c_h+h-1}(1-\theta_1)^{c_t+t-1}\end{aligned}$$

$$logp(\theta_1|\mathcal{D}) = (c_h+h-1)log(\theta_1) + (c_t+t-1)log(1-\theta_1)$$

By taking derivatives w.r.t θ_1, we get:

$$\hat{\theta}_1 = \frac{c_h+h-1}{c_h+c_t+h+t-2}$$

And that,

$$\hat{\theta}_2 = \frac{n_h}{n_h+n_t}\frac{1}{\theta_1}$$