



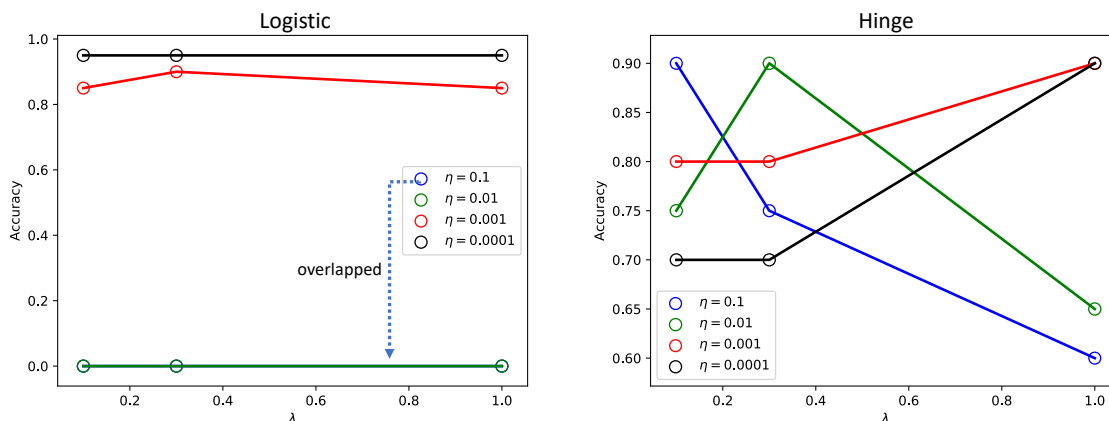
CSE 5523 HW4: Logistic Regression and Hinge Loss

1) Logistic loss and Hinge Loss formalism are as follows:

$$\mathcal{L}_{\log}(w) = \sum_l \log(1 + \exp(-y_l w \cdot x_l)) + \lambda \sum_i w_i^2$$

$$\mathcal{L}_{\text{hinge}}(w) = \sum_l \max(0, 1 - y_l w \cdot x_l) + \lambda \sum_i w_i^2$$

Following are the results for both SgdHinge and SgdLogistic cost functions. The results show that the choice of  $\lambda = 1$  and  $\eta = 0.0001$  gives the highest accuracy among other parameters for both Hinge and Logistic algorithms. So, I proceed with these set of hyper parameters.



**a) Loss for first and last iteration for each cross-validation set: Hinge Loss method:**

CROSS VALIDATION 0

Iteration %s HINGE-LOSS %s (0, 53.0)

Iteration %s HINGE-LOSS %s (99, 0.03614375154312088)

[-7.19350667e-04 5.28731121e-05 8.03733400e-05 ... 2.21954817e-04  
 2.93276678e-04 -3.52224393e-05]

CROSS VALIDATION 1

Iteration %s HINGE-LOSS %s (0, 53.0)

Iteration %s HINGE-LOSS %s (99, 0.09099633147113441)

[-9.39498318e-04 7.45379140e-05 -3.62801387e-04 ... 1.06801468e-03  
 4.06331795e-04 1.73370619e-04]



---

CROSS VALIDATION 2

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.045894838022805196)  
[-4.48692799e-04 -1.20252892e-05 -2.92905430e-04 ... 7.58254788e-04  
2.90507161e-04 9.86985257e-05]

CROSS VALIDATION 3

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04116822677285094)  
[-1.02908247e-03 -1.03757346e-04 -1.84246388e-04 ... 7.82692432e-04  
4.72304691e-04 7.07077384e-05]

CROSS VALIDATION 4

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04116822677285094)  
[-1.02908247e-03 -1.03757346e-04 -1.84246388e-04 ... 7.82692432e-04  
4.72304691e-04 7.07077384e-05]

CROSS VALIDATION 5

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.03737833312208725)  
[-3.54146849e-04 2.83508198e-04 -3.34655430e-04 ... 7.17442851e-05  
9.37207667e-05 2.08685346e-04]

CROSS VALIDATION 6

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.09979629121530945)  
[-0.00063591 0.00037123 -0.00022731 ... 0.00098958 0.00074402  
0.00022141]

CROSS VALIDATION 7

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.041169886390632585)  
[-1.02908560e-03 -1.03732389e-04 -1.84231576e-04 ... 7.82732225e-04  
4.72350742e-04 7.07079639e-05]

CROSS VALIDATION 8

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04131944553270079)  
[-7.94933003e-04 1.94359872e-04 -5.13605043e-04 ... 6.34901754e-04  
-6.85688922e-05 3.30197941e-05]

CROSS VALIDATION 9

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04032696389276679)  
[-1.04788956e-03 -9.61675301e-05 -2.96710925e-04 ... 7.17600805e-04  
5.68616474e-04 6.97473877e-05]

CROSS VALIDATION 10

Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04343144996425163)  
[-0.0007716 -0.00014186 -0.00055766 ... 0.00049646 0.00016733]



---

0.00010442]  
CROSS VALIDATION 11  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04117182628772593)  
[-1.02912840e-03 -1.03762318e-04 -1.84274851e-04 ... 7.82737689e-04  
4.72386551e-04 7.07079612e-05]  
CROSS VALIDATION 12  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.044147538269705154)  
[-1.54691954e-03 -1.97480050e-04 -6.40781005e-04 ... 7.28231770e-04  
4.41491697e-04 3.53283939e-05]  
CROSS VALIDATION 13  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04117250702981224)  
[-1.02908954e-03 -1.03744436e-04 -1.84274485e-04 ... 7.82737674e-04  
4.72383813e-04 7.07149076e-05]  
CROSS VALIDATION 14  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.045345766862434335)  
[-9.37497666e-04 -4.40566229e-05 -2.29380638e-04 ... 5.91316523e-04  
5.77299626e-04 6.55305881e-05]  
CROSS VALIDATION 15  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04117332837652634)  
[-1.02913461e-03 -1.03766262e-04 -1.84294264e-04 ... 7.82768197e-04  
4.72414297e-04 7.07220057e-05]  
CROSS VALIDATION 16  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 2.649885140131366)  
[-0.00227171 -0.00044884 -0.00125121 ... 0.00169382 0.00097931  
0.00023254]  
CROSS VALIDATION 17  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.041174068207124104)  
[-1.02916377e-03 -1.03767962e-04 -1.84303540e-04 ... 7.82769239e-04  
4.72413996e-04 7.07149796e-05]  
CROSS VALIDATION 18  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.04301965519501956)  
[-1.37217881e-03 -2.23605239e-04 -8.05497092e-04 ... 9.49550960e-04  
4.60272709e-04 6.94953685e-05]  
CROSS VALIDATION 19  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (99, 0.03490415916340902)



---

[-7.38725931e-04 -2.15720788e-04 -5.60145378e-04 ... 3.83434016e-04  
1.30854810e-04 6.93243957e-05]  
Accuracy (Hinge Loss): 0.9

**b) Loss for first and last iteration for each cross-validation set: Logistic  
Loss method:**

CROSS VALIDATION 0

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.036295850447107876

[-1.75691859e-04 2.42480868e-04 8.49903325e-05 ... 2.24590483e-04  
1.86298667e-05 1.09732468e-04]

CROSS VALIDATION 1

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.03519230359530414

[-4.18643517e-04 1.47717833e-04 -1.68762327e-04 ... 1.22388490e-04  
-4.88964670e-05 1.45264756e-04]

CROSS VALIDATION 2

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.03605001637674191

[-0.00060005 -0.00022885 -0.00028811 ... 0.00032985 0.00020163  
0.0001104]

CROSS VALIDATION 3

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.03515155825246683

[-4.26604455e-04 1.69986916e-04 -1.66226111e-04 ... 1.36020105e-04  
-4.30481278e-05 1.47888934e-04]

CROSS VALIDATION 4

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.03511522471782297

[-4.15718609e-04 1.75806337e-04 -1.62317958e-04 ... 1.33254841e-04  
-3.95627398e-05 1.48965165e-04]

CROSS VALIDATION 5

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.035379667679712234

[-3.21742332e-04 2.17931071e-04 -1.63539347e-04 ... 6.68397546e-05  
-3.95099203e-05 1.78657037e-04]

CROSS VALIDATION 6

Iteration 0 LOGISTIC LOSS: 36.73680056967707

Iteration 99 LOGISTIC LOSS: 0.03515597927742838

[-4.24501365e-04 1.71308011e-04 -1.61583775e-04 ... 1.28532682e-04  
-4.38321682e-05 1.47171415e-04]

CROSS VALIDATION 7

Iteration 0 LOGISTIC LOSS: 36.73680056967707



---

Iteration 99 LOGISTIC LOSS: 0.035097200594500146  
[-4.30071222e-04 1.70483725e-04 -1.67495461e-04 ... 1.25796296e-04  
-4.61596776e-05 1.49472066e-04]  
CROSS VALIDATION 8  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.035121855859256174  
[-4.16561807e-04 1.76262928e-04 -1.49587294e-04 ... 1.30587005e-04  
-4.97631823e-05 1.46470267e-04]  
CROSS VALIDATION 9  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.037412447944346686  
[-3.83137533e-04 2.02460620e-04 -1.12092875e-04 ... 2.57630342e-04  
-3.30756685e-05 1.43159658e-04]  
CROSS VALIDATION 10  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.03960968070085072  
[-3.89714754e-04 1.66322229e-04 -1.80285245e-04 ... 2.45657513e-04  
-1.57387758e-05 1.53089928e-04]  
CROSS VALIDATION 11  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.034991275358579654  
[-4.21005626e-04 1.78048800e-04 -1.51627421e-04 ... 1.30642137e-04  
-4.35996267e-05 1.47142149e-04]  
CROSS VALIDATION 12  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.03222128177479262  
[-7.32208821e-04 -3.78860179e-05 -2.25834036e-04 ... 6.97550916e-05  
-4.89209248e-05 8.52395147e-05]  
CROSS VALIDATION 13  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.037487429290994315  
[-5.81883968e-04 -1.15178216e-04 -2.67993325e-04 ... 7.19849931e-05  
3.29708354e-05 1.29757704e-04]  
CROSS VALIDATION 14  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.03492813329209701  
[-3.95703504e-04 1.87212067e-04 -1.48549887e-04 ... 1.13544561e-04  
-5.90732748e-05 1.44525656e-04]  
CROSS VALIDATION 15  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.035080382899578884  
[-4.21454549e-04 1.85234444e-04 -1.60409216e-04 ... 1.31181151e-04  
-4.45761022e-05 1.46760651e-04]  
CROSS VALIDATION 16



---

Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.034666879941777445  
[-4.03596668e-04 1.75529651e-04 -1.53428418e-04 ... 1.27882399e-04  
-4.54109117e-05 1.51969466e-04]  
CROSS VALIDATION 17  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.035220345998890006  
[-4.19817472e-04 1.77275221e-04 -1.55593193e-04 ... 1.33159786e-04  
-4.70041035e-05 1.47062048e-04]  
CROSS VALIDATION 18  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.03796614630042348  
[-3.53617763e-04 -1.22877250e-04 -5.15969351e-04 ... 3.69116686e-04  
-1.58131737e-05 1.27083188e-04]  
CROSS VALIDATION 19  
Iteration 0 LOGISTIC LOSS: 36.73680056967707  
Iteration 99 LOGISTIC LOSS: 0.03448265038497007  
[-4.82173816e-04 1.76352418e-04 -1.93666351e-04 ... 1.15707333e-04  
-5.10882434e-05 1.56326527e-04]  
Accuracy (Logistic Loss): 0.95

**c) Hinge loss for all iterations for Last Cross Validation (19):**

CROSS VALIDATION 19  
Iteration %s HINGE-LOSS %s (0, 53.0)  
Iteration %s HINGE-LOSS %s (1, 30.47091981078378)  
Iteration %s HINGE-LOSS %s (2, 14.00825596949709)  
Iteration %s HINGE-LOSS %s (3, 0.2672032334927989)  
Iteration %s HINGE-LOSS %s (4, 0.26159759410692107)  
Iteration %s HINGE-LOSS %s (5, 0.25610955506784205)  
Iteration %s HINGE-LOSS %s (6, 0.2507366492454781)  
Iteration %s HINGE-LOSS %s (7, 0.24547646126750822)  
Iteration %s HINGE-LOSS %s (8, 0.24032662643355165)  
Iteration %s HINGE-LOSS %s (9, 0.23528482965212402)  
Iteration %s HINGE-LOSS %s (10, 0.23034880439989575)  
Iteration %s HINGE-LOSS %s (11, 0.22551633170278398)  
Iteration %s HINGE-LOSS %s (12, 0.22078523913841996)  
Iteration %s HINGE-LOSS %s (13, 0.21615339985954335)  
Iteration %s HINGE-LOSS %s (14, 0.2116187316378853)  
Iteration %s HINGE-LOSS %s (15, 0.20717919592810954)  
Iteration %s HINGE-LOSS %s (16, 0.2028327969513906)  
Iteration %s HINGE-LOSS %s (17, 0.19857758079821813)  
Iteration %s HINGE-LOSS %s (18, 0.19441163455002336)





---

Iteration %s HINGE-LOSS %s (19, 0.1903330854192327)  
Iteration %s HINGE-LOSS %s (20, 0.18634009990736225)  
Iteration %s HINGE-LOSS %s (21, 0.18243088298077414)  
Iteration %s HINGE-LOSS %s (22, 0.17860367726372564)  
Iteration %s HINGE-LOSS %s (23, 0.17485676224834604)  
Iteration %s HINGE-LOSS %s (24, 0.1711884535211883)  
Iteration %s HINGE-LOSS %s (25, 0.16759710200600633)  
Iteration %s HINGE-LOSS %s (26, 0.1640810932224181)  
Iteration %s HINGE-LOSS %s (27, 0.1606388465601214)  
Iteration %s HINGE-LOSS %s (28, 0.15726881456833533)  
Iteration %s HINGE-LOSS %s (29, 0.15396948226014934)  
Iteration %s HINGE-LOSS %s (30, 0.15073936643146502)  
Iteration %s HINGE-LOSS %s (31, 0.14757701499422746)  
Iteration %s HINGE-LOSS %s (32, 0.1444810063236429)  
Iteration %s HINGE-LOSS %s (33, 0.1414499486190927)  
Iteration %s HINGE-LOSS %s (34, 0.13848247927845336)  
Iteration %s HINGE-LOSS %s (35, 0.13557726428554331)  
Iteration %s HINGE-LOSS %s (36, 0.13273299761042062)  
Iteration %s HINGE-LOSS %s (37, 0.12994840062226085)  
Iteration %s HINGE-LOSS %s (38, 0.12722222151455329)  
Iteration %s HINGE-LOSS %s (39, 0.12455323474235515)  
Iteration %s HINGE-LOSS %s (40, 0.1219402404713519)  
Iteration %s HINGE-LOSS %s (41, 0.11938206403847555)  
Iteration %s HINGE-LOSS %s (42, 0.11687755542383897)  
Iteration %s HINGE-LOSS %s (43, 0.11442558873374782)  
Iteration %s HINGE-LOSS %s (44, 0.11202506169455899)  
Iteration %s HINGE-LOSS %s (45, 0.10967489515715692)  
Iteration %s HINGE-LOSS %s (46, 0.10737403261182567)  
Iteration %s HINGE-LOSS %s (47, 0.1051214397132985)  
Iteration %s HINGE-LOSS %s (48, 0.10291610381577115)  
Iteration %s HINGE-LOSS %s (49, 0.10075703351767043)  
Iteration %s HINGE-LOSS %s (50, 0.09864325821597264)  
Iteration %s HINGE-LOSS %s (51, 0.0965738276698723)  
Iteration %s HINGE-LOSS %s (52, 0.09454781157360448)  
Iteration %s HINGE-LOSS %s (53, 0.09256429913822886)  
Iteration %s HINGE-LOSS %s (54, 0.09062239868218734)  
Iteration %s HINGE-LOSS %s (55, 0.08872123723045185)  
Iteration %s HINGE-LOSS %s (56, 0.08685996012208096)  
Iteration %s HINGE-LOSS %s (57, 0.08503773062600997)  
Iteration %s HINGE-LOSS %s (58, 0.08325372956490129)  
Iteration %s HINGE-LOSS %s (59, 0.08150715494688562)  
Iteration %s HINGE-LOSS %s (60, 0.07979722160502947)  
Iteration %s HINGE-LOSS %s (61, 0.0781231608443657)  
Iteration %s HINGE-LOSS %s (62, 0.0764842200963293)



---

Iteration %s HINGE-LOSS %s (63, 0.07487966258044246)  
Iteration %s HINGE-LOSS %s (64, 0.07330876697309764)  
Iteration %s HINGE-LOSS %s (65, 0.07177082708328861)  
Iteration %s HINGE-LOSS %s (66, 0.07026515153514463)  
Iteration %s HINGE-LOSS %s (67, 0.0687910634571248)  
Iteration %s HINGE-LOSS %s (68, 0.06734790017773258)  
Iteration %s HINGE-LOSS %s (69, 0.06593501292761386)  
Iteration %s HINGE-LOSS %s (70, 0.06455176654790509)  
Iteration %s HINGE-LOSS %s (71, 0.0631975392046994)  
Iteration %s HINGE-LOSS %s (72, 0.06187172210950333)  
Iteration %s HINGE-LOSS %s (73, 0.06057371924555794)  
Iteration %s HINGE-LOSS %s (74, 0.05930294709990141)  
Iteration %s HINGE-LOSS %s (75, 0.05805883440105266)  
Iteration %s HINGE-LOSS %s (76, 0.05684082186219816)  
Iteration %s HINGE-LOSS %s (77, 0.055648361929766264)  
Iteration %s HINGE-LOSS %s (78, 0.054480918537276456)  
Iteration %s HINGE-LOSS %s (79, 0.053337966864352215)  
Iteration %s HINGE-LOSS %s (80, 0.052218993100789896)  
Iteration %s HINGE-LOSS %s (81, 0.0511234942155769)  
Iteration %s HINGE-LOSS %s (82, 0.05005097773075579)  
Iteration %s HINGE-LOSS %s (83, 0.049000961500032365)  
Iteration %s HINGE-LOSS %s (84, 0.047972973492028455)  
Iteration %s HINGE-LOSS %s (85, 0.046966551578081696)  
Iteration %s HINGE-LOSS %s (86, 0.04598124332449706)  
Iteration %s HINGE-LOSS %s (87, 0.04501660578915684)  
Iteration %s HINGE-LOSS %s (88, 0.04407220532239739)  
Iteration %s HINGE-LOSS %s (89, 0.043147617372063386)  
Iteration %s HINGE-LOSS %s (90, 0.04224242629265176)  
Iteration %s HINGE-LOSS %s (91, 0.0413562251584596)  
Iteration %s HINGE-LOSS %s (92, 0.04048861558065209)  
Iteration %s HINGE-LOSS %s (93, 0.039639207528168)  
Iteration %s HINGE-LOSS %s (94, 0.0388076191523826)  
Iteration %s HINGE-LOSS %s (95, 0.037993476615448794)  
Iteration %s HINGE-LOSS %s (96, 0.03719641392223955)  
Iteration %s HINGE-LOSS %s (97, 0.03641607275581598)  
Iteration %s HINGE-LOSS %s (98, 0.03565210231634712)  
Iteration %s HINGE-LOSS %s (99, 0.03490415916340902)  
[-7.38725931e-04 -2.15720788e-04 -5.60145378e-04 ... 3.83434016e-04  
1.30854810e-04 6.93243957e-05]

**d) Logistic loss for all iterations for Last Cross Validation (19):**

CROSS VALIDATION 19  
Iteration 0 LOGISTIC LOSS: 36.73680056967707





---

Iteration 1 LOGISTIC LOSS: 7.8550761951967445  
Iteration 2 LOGISTIC LOSS: 0.20669914772699016  
Iteration 3 LOGISTIC LOSS: 0.18904262702795874  
Iteration 4 LOGISTIC LOSS: 0.1847139903488153  
Iteration 5 LOGISTIC LOSS: 0.18062011689744892  
Iteration 6 LOGISTIC LOSS: 0.17669966472024573  
Iteration 7 LOGISTIC LOSS: 0.1729195738490897  
Iteration 8 LOGISTIC LOSS: 0.16925988220298194  
Iteration 9 LOGISTIC LOSS: 0.16570747329879548  
Iteration 10 LOGISTIC LOSS: 0.16225313246690845  
Iteration 11 LOGISTIC LOSS: 0.15889002064592872  
Iteration 12 LOGISTIC LOSS: 0.15561282322072556  
Iteration 13 LOGISTIC LOSS: 0.15241724734351006  
Iteration 14 LOGISTIC LOSS: 0.14929971103962705  
Iteration 15 LOGISTIC LOSS: 0.14625714348859956  
Iteration 16 LOGISTIC LOSS: 0.1432868525729674  
Iteration 17 LOGISTIC LOSS: 0.14038643460033934  
Iteration 18 LOGISTIC LOSS: 0.1375537112524283  
Iteration 19 LOGISTIC LOSS: 0.1347866845350753  
Iteration 20 LOGISTIC LOSS: 0.13208350385499462  
Iteration 21 LOGISTIC LOSS: 0.12944244138367766  
Iteration 22 LOGISTIC LOSS: 0.1268618731457539  
Iteration 23 LOGISTIC LOSS: 0.12434026409655974  
Iteration 24 LOGISTIC LOSS: 0.12187615600648012  
Iteration 25 LOGISTIC LOSS: 0.11946815734942623  
Iteration 26 LOGISTIC LOSS: 0.11711493465978604  
Iteration 27 LOGISTIC LOSS: 0.11481520501231209  
Iteration 28 LOGISTIC LOSS: 0.11256772941488957  
Iteration 29 LOGISTIC LOSS: 0.1103713069986774  
Iteration 30 LOGISTIC LOSS: 0.1082247699527714  
Iteration 31 LOGISTIC LOSS: 0.10612697918786115  
Iteration 32 LOGISTIC LOSS: 0.10407682073015868  
Iteration 33 LOGISTIC LOSS: 0.10207320284788997  
Iteration 34 LOGISTIC LOSS: 0.10011505390188317  
Iteration 35 LOGISTIC LOSS: 0.09820132089344619  
Iteration 36 LOGISTIC LOSS: 0.0963309686607872  
Iteration 37 LOGISTIC LOSS: 0.0945029796532306  
Iteration 38 LOGISTIC LOSS: 0.09271635419363143  
Iteration 39 LOGISTIC LOSS: 0.09097011112588176  
Iteration 40 LOGISTIC LOSS: 0.08926328873779257  
Iteration 41 LOGISTIC LOSS: 0.08759494585018901  
Iteration 42 LOGISTIC LOSS: 0.08596416297036492  
Iteration 43 LOGISTIC LOSS: 0.08437004342082553  
Iteration 44 LOGISTIC LOSS: 0.08281171437051839



---

Iteration 45 LOGISTIC LOSS: 0.08128832771373433  
Iteration 46 LOGISTIC LOSS: 0.07979906075937397  
Iteration 47 LOGISTIC LOSS: 0.07834311670884211  
Iteration 48 LOGISTIC LOSS: 0.07691972491332623  
Iteration 49 LOGISTIC LOSS: 0.07552814091008818  
Iteration 50 LOGISTIC LOSS: 0.07416764624275758  
Iteration 51 LOGISTIC LOSS: 0.07283754807295413  
Iteration 52 LOGISTIC LOSS: 0.07153717859068325  
Iteration 53 LOGISTIC LOSS: 0.07026589422993211  
Iteration 54 LOGISTIC LOSS: 0.06902307469460384  
Iteration 55 LOGISTIC LOSS: 0.06780812179929367  
Iteration 56 LOGISTIC LOSS: 0.06662045813005622  
Iteration 57 LOGISTIC LOSS: 0.06545952553253383  
Iteration 58 LOGISTIC LOSS: 0.06432478343875571  
Iteration 59 LOGISTIC LOSS: 0.06321570704943721  
Iteration 60 LOGISTIC LOSS: 0.06213178539521606  
Iteration 61 LOGISTIC LOSS: 0.06107251930761836  
Iteration 62 LOGISTIC LOSS: 0.06003741933787785  
Iteration 63 LOGISTIC LOSS: 0.05902600366849225  
Iteration 64 LOGISTIC LOSS: 0.05803779606794625  
Iteration 65 LOGISTIC LOSS: 0.05707232394273044  
Iteration 66 LOGISTIC LOSS: 0.056129116542242674  
Iteration 67 LOGISTIC LOSS: 0.05520770337103631  
Iteration 68 LOGISTIC LOSS: 0.054307612858953754  
Iteration 69 LOGISTIC LOSS: 0.05342837133307558  
Iteration 70 LOGISTIC LOSS: 0.05256950232624602  
Iteration 71 LOGISTIC LOSS: 0.051730526245788125  
Iteration 72 LOGISTIC LOSS: 0.05091096041322576  
Iteration 73 LOGISTIC LOSS: 0.05011031947238919  
Iteration 74 LOGISTIC LOSS: 0.049328116149775406  
Iteration 75 LOGISTIC LOSS: 0.04856386233827226  
Iteration 76 LOGISTIC LOSS: 0.0478170704642222  
Iteration 77 LOGISTIC LOSS: 0.04708725508860277  
Iteration 78 LOGISTIC LOSS: 0.046373934686602704  
Iteration 79 LOGISTIC LOSS: 0.04567663354598721  
Iteration 80 LOGISTIC LOSS: 0.044994883723631834  
Iteration 81 LOGISTIC LOSS: 0.04432822700114811  
Iteration 82 LOGISTIC LOSS: 0.04367621678442517  
Iteration 83 LOGISTIC LOSS: 0.043038419897626815  
Iteration 84 LOGISTIC LOSS: 0.042414418229373044  
Iteration 85 LOGISTIC LOSS: 0.04180381019687046  
Iteration 86 LOGISTIC LOSS: 0.041206212002213354  
Iteration 87 LOGISTIC LOSS: 0.04062125866356496  
Iteration 88 LOGISTIC LOSS: 0.040048604811996125



---

```
Iteration 89 LOGISTIC LOSS: 0.03948792525216665
Iteration 90 LOGISTIC LOSS: 0.03893891529160108
Iteration 91 LOGISTIC LOSS: 0.03840129084879571
Iteration 92 LOGISTIC LOSS: 0.037874788354877106
Iteration 93 LOGISTIC LOSS: 0.03735916446683397
Iteration 94 LOGISTIC LOSS: 0.036854195612678047
Iteration 95 LOGISTIC LOSS: 0.03635967739018882
Iteration 96 LOGISTIC LOSS: 0.035875423841402895
Iteration 97 LOGISTIC LOSS: 0.035401266624674556
Iteration 98 LOGISTIC LOSS: 0.034937054105386785
Iteration 99 LOGISTIC LOSS: 0.03448265038497007
[-4.82173816e-04 1.76352418e-04 -1.93666351e-04 ... 1.15707333e-04
-5.10882434e-05 1.56326527e-04]
```

- 2) I used the optimum  $\lambda = 1$  and  $\eta = 0.0001$  parameters to test the data. The accuracy on the test data are as follows. The interesting finding is that if **MAXiter** is large enough, both Hinge and Logistic loss converge to the higher and very similar accuracy. For  $\text{MAXiter} \sim 100$  and  $\text{abs}(\text{previous\_loss} - \text{Current\_loss}) \sim 0.0001$ , the results are as follows:

```
Accuracy (Hinge Loss):    0.8529411764705882
Accuracy (Logistic Loss): 0.8529411764705882
```

If I used lower MAXiter, the second condition which is the loss criterion governs the iteration loop. Using  $\text{MAXiter} \sim 10$  and  $\text{abs}(\text{previous\_loss} - \text{Current\_loss}) \sim 0.0001$ , result accuracy is as follows:

```
Accuracy (Hinge Loss):    0.8529411764705882
Accuracy (Logistic Loss): 0.8235294117647058
```

- 3) Bonus: Inspecting Model parameter

To find which brain regions seem to be the most important for the predictions, the weights are collapsed after each cross-validation iteration. For each **rois** array, I calculated the average weight of corresponding column in the collapsed weight vector. The code is added to the cross-validation function in the attached .py file. The result for a sample from Logistic loss is as follows:

CrossValidation 19:

```
[-4.42781457e-04 2.45333669e-04 -1.61192144e-04 ... 1.30569778e-04
9.66993185e-06 1.72477901e-04]
```



```
[W_rois_element(Weight=array(['CALC'], dtype='<U4'),
Name=0.01774701129913674), W_rois_element(Weight=array(['LDLPFC'],
dtype='<U6'), Name=0.02178480901098556),
W_rois_element(Weight=array(['LFEF'], dtype='<U4'),
Name=0.005321089703719855), W_rois_element(Weight=array(['LIFG'],
dtype='<U4'), Name=0.012721883271537488),
W_rois_element(Weight=array(['LIPL'], dtype='<U4'),
Name=0.006386507089889357), W_rois_element(Weight=array(['LIPS'],
dtype='<U4'), Name=0.010648286487784328),
W_rois_element(Weight=array(['LIT'], dtype='<U3'),
Name=0.013004844122790722), W_rois_element(Weight=array(['LOPER'],
dtype='<U5'), Name=0.007869788356925218),
W_rois_element(Weight=array(['LPPREC'], dtype='<U6'),
Name=0.006498463978624484), W_rois_element(Weight=array(['LSGA'],
dtype='<U4'), Name=0.0002617551884512115),
W_rois_element(Weight=array(['LSPL'], dtype='<U4'),
Name=0.01618690918511231), W_rois_element(Weight=array(['LT'],
dtype='<U2'), Name=0.015982805397797274),
W_rois_element(Weight=array(['LTRIA'], dtype='<U5'),
Name=0.004852094914612262), W_rois_element(Weight=array(['RDL PFC'],
dtype='<U6'), Name=0.020267838483047414),
W_rois_element(Weight=array(['RFEF'], dtype='<U4'),
Name=0.0037385341708836527), W_rois_element(Weight=array(['RIPL'],
dtype='<U4'), Name=0.0072936968869019465),
W_rois_element(Weight=array(['RIPS'], dtype='<U4'),
Name=0.009621893018209272), W_rois_element(Weight=array(['RIT'],
dtype='<U3'), Name=0.014783267738311071),
W_rois_element(Weight=array(['ROPER'], dtype='<U5'),
Name=0.0092352325850264), W_rois_element(Weight=array(['RPPREC'],
dtype='<U6'), Name=0.006586498290062181),
W_rois_element(Weight=array(['RSGA'], dtype='<U4'),
Name=0.0027629597870617866), W_rois_element(Weight=array(['RSPL'],
dtype='<U4'), Name=0.011966754370123234),
W_rois_element(Weight=array(['RT'], dtype='<U2'),
Name=0.01433306662873187), W_rois_element(Weight=array(['RTRIA'],
dtype='<U5'), Name=0.0026291396211056),
W_rois_element(Weight=array(['SMA'], dtype='<U3'),
Name=0.009338992439223743)]
```

From the above information, it seems that LDLPFC, LSPL, LT and RDL PFC are among the highest weight values through the cross validation.