

# Logistic Lasso optimization

Hun, Jimmy

3/19/2022

```
cancer_df <- read.csv("~/Downloads/breast-cancer.csv") %>% janitor::clean_names()

data <-
  cancer_df %>% dplyr::select(-id, -x) %>%
  mutate(diagnosis = ifelse(diagnosis == "M", 1, 0)) %>% distinct()

set.seed(1)
split <- initial_split(data, prop = 0.8)

training_df <- split %>% training()

testing_df <- split %>% testing()
```

## Standardization

```
for (i in 2:length(training_df)) {
  training_df[,i] = (training_df[,i] - mean(training_df[,i]))/sd(training_df[,i])
}
```

## Checking standardization

```
summary <- skimr::skim_without_charts(training_df) %>% data.frame()
summary[1:6] %>% kable()
```

skim_type	skim_variable	n_missing	complete_rate	numeric.mean	numeric.sd
numeric	diagnosis	0	1	0.3714286	0.4837186
numeric	radius_mean	0	1	0.0000000	1.0000000
numeric	texture_mean	0	1	0.0000000	1.0000000
numeric	perimeter_mean	0	1	0.0000000	1.0000000
numeric	area_mean	0	1	0.0000000	1.0000000
numeric	smoothness_mean	0	1	0.0000000	1.0000000
numeric	compactness_mean	0	1	0.0000000	1.0000000
numeric	concavity_mean	0	1	0.0000000	1.0000000
numeric	concave_points_mean	0	1	0.0000000	1.0000000
numeric	symmetry_mean	0	1	0.0000000	1.0000000
numeric	fractal_dimension_mean	0	1	0.0000000	1.0000000
numeric	radius_se	0	1	0.0000000	1.0000000
numeric	texture_se	0	1	0.0000000	1.0000000
numeric	perimeter_se	0	1	0.0000000	1.0000000
numeric	area_se	0	1	0.0000000	1.0000000
numeric	smoothness_se	0	1	0.0000000	1.0000000
numeric	compactness_se	0	1	0.0000000	1.0000000
numeric	concavity_se	0	1	0.0000000	1.0000000
numeric	concave_points_se	0	1	0.0000000	1.0000000
numeric	symmetry_se	0	1	0.0000000	1.0000000
numeric	fractal_dimension_se	0	1	0.0000000	1.0000000
numeric	radius_worst	0	1	0.0000000	1.0000000
numeric	texture_worst	0	1	0.0000000	1.0000000
numeric	perimeter_worst	0	1	0.0000000	1.0000000
numeric	area_worst	0	1	0.0000000	1.0000000
numeric	smoothness_worst	0	1	0.0000000	1.0000000
numeric	compactness_worst	0	1	0.0000000	1.0000000
numeric	concavity_worst	0	1	0.0000000	1.0000000
numeric	concave_points_worst	0	1	0.0000000	1.0000000
numeric	symmetry_worst	0	1	0.0000000	1.0000000
numeric	fractal_dimension_worst	0	1	0.0000000	1.0000000

```
X <- training_df %>% dplyr::select(-diagnosis) %>% as.matrix()
y <- training_df$diagnosis %>% as.matrix()

dim(X)
```

```
## [1] 455 30
```

```
dim(y)
```

```
## [1] 455 1
```

## Helper Functions

```
# logistic function
logistic <- function(x) 1 / (1 + exp(-x))

# soft threshold
```

```

S <- function(beta, gamma) {
  if (beta > 0 && gamma < abs(beta)) {
    beta - gamma
  } else if (beta < 0 && gamma < abs(beta)) {
    beta + gamma
  } else {
    0
  }
}

# probability adjustment function
p_adj <- function(p, epsilon) {
  if (p < epsilon) {
    0
  } else if (p > 1 - epsilon) {
    1
  } else {
    p
  }
}

# weight adjustment function
w_adj <- function(p, epsilon) {
  if ((p < epsilon) | (p > 1 - epsilon)) {
    epsilon
  } else {
    p * (1 - p)
  }
}

```

Computing lambda max in two ways and defining lambda sequence

```

alpha = 1
n = length(y)

lambda_max <- max(abs(t(X) %*% y))/n

same_lambda_max <- max(abs(t(y - mean(y)*(1 - mean(y))) %*% X ))/(alpha*n)

lambda_max

## [1] 0.386795

same_lambda_max

## [1] 0.386795

lambda_max <- lambda_max %>% round(digits = 2)

```

```
epsilon = 1e-5
lambda_seq <- seq(lambda_max, 1e-5*lambda_max, length = 100)
```

## Toy example

```
set.seed(1)

n <- 1000
X <- scale(matrix(rnorm(3 * n), c(n, 3)))
X <- as.matrix(cbind(rep(1, n), X))
y <- 1 * (runif(n) > 0.5)

lambda_max <- max(abs(t(X) %*% y))/n %>% round(digits = 2)
epsilon <- 1e-5

lambda <- lambda_max

lambda_seq <- seq(lambda_max, epsilon*lambda_max, length = 100)

# initialize parameters
beta <- list()
beta_old <- list()
b_k_temp <- list()

# creating initial beta list
for (i in 1:length(lambda_seq)) {
  beta[[i]] <- rep(0, ncol(X))
}

# loop for decrement lambda
for (i in 1:length(lambda_seq)) {
  p <- map_dbl(logistic(X %*% beta[[i]]), p_adj, epsilon)
  w <- map_dbl(p, w_adj, epsilon)
  z <- X %*% beta[[i]] + (y - p) / w

  terminate <- 0
  iter <- 1
  while (terminate < 1) {

    beta_old[[i]] <- beta[[i]]

    for (k in 1:ncol(X)) {
      x_k <- X[, k]
      x_notk <- X[, -k]
      b_notk <- beta[[i]][-k]

      # un-penalized coefficient update
      b_k_temp <- sum(w * (z - x_notk %*% b_notk) * x_k) / sum(w * x_k^2)
      # shrinkage update
      b_k <- S(b_k_temp, lambda * (k > 1) / mean(w * x_k^2))
    }
  }
}
```

```

#b_k      <- S(sum(w*x_k*(z - x_notk %*% b_notk)), lambda) / sum(w * x_k^2)
# update beta list along with other parameters
beta[[i]][k] <- b_k

iter <- iter + 1
lambda <- lambda_seq[i]

}
if (iter == 200 | max(abs(beta[[i]] - beta_old[[i]])) < 1e-10) {
  terminate <- 1
}
}
}

# Estimates from Coordinate Descent
beta <- data.frame(beta)

for (i in 1:length(lambda_seq)) {
  colnames(beta)[i] <- paste0(lambda_seq[i] %>% round(digits = 6))
}

beta <- t(beta) %>% data.frame()

beta %>% mutate(lambda = rownames(beta)) %>% relocate(lambda) %>% `rownames<-`( NULL )

```

##	lambda	X1	X2	X3	X4
## 1	0.526	0.104	0.00000000	0.00000000	0.00000000
## 2	0.520687	0.104	0.00000000	0.00000000	0.00000000
## 3	0.515374	0.104	0.00000000	0.00000000	0.00000000
## 4	0.510061	0.104	0.00000000	0.00000000	0.00000000
## 5	0.504748	0.104	0.00000000	0.00000000	0.00000000
## 6	0.499435	0.104	0.00000000	0.00000000	0.00000000
## 7	0.494122	0.104	0.00000000	0.00000000	0.00000000
## 8	0.488808	0.104	0.00000000	0.00000000	0.00000000
## 9	0.483495	0.104	0.00000000	0.00000000	0.00000000
## 10	0.478182	0.104	0.00000000	0.00000000	0.00000000
## 11	0.472869	0.104	0.00000000	0.00000000	0.00000000
## 12	0.467556	0.104	0.00000000	0.00000000	0.00000000
## 13	0.462243	0.104	0.00000000	0.00000000	0.00000000
## 14	0.45693	0.104	0.00000000	0.00000000	0.00000000
## 15	0.451617	0.104	0.00000000	0.00000000	0.00000000
## 16	0.446304	0.104	0.00000000	0.00000000	0.00000000
## 17	0.440991	0.104	0.00000000	0.00000000	0.00000000
## 18	0.435678	0.104	0.00000000	0.00000000	0.00000000
## 19	0.430365	0.104	0.00000000	0.00000000	0.00000000
## 20	0.425052	0.104	0.00000000	0.00000000	0.00000000
## 21	0.419738	0.104	0.00000000	0.00000000	0.00000000
## 22	0.414425	0.104	0.00000000	0.00000000	0.00000000
## 23	0.409112	0.104	0.00000000	0.00000000	0.00000000
## 24	0.403799	0.104	0.00000000	0.00000000	0.00000000
## 25	0.398486	0.104	0.00000000	0.00000000	0.00000000

## 26	0.393173	0.104	0.00000000	0.00000000	0.00000000
## 27	0.38786	0.104	0.00000000	0.00000000	0.00000000
## 28	0.382547	0.104	0.00000000	0.00000000	0.00000000
## 29	0.377234	0.104	0.00000000	0.00000000	0.00000000
## 30	0.371921	0.104	0.00000000	0.00000000	0.00000000
## 31	0.366608	0.104	0.00000000	0.00000000	0.00000000
## 32	0.361295	0.104	0.00000000	0.00000000	0.00000000
## 33	0.355981	0.104	0.00000000	0.00000000	0.00000000
## 34	0.350668	0.104	0.00000000	0.00000000	0.00000000
## 35	0.345355	0.104	0.00000000	0.00000000	0.00000000
## 36	0.340042	0.104	0.00000000	0.00000000	0.00000000
## 37	0.334729	0.104	0.00000000	0.00000000	0.00000000
## 38	0.329416	0.104	0.00000000	0.00000000	0.00000000
## 39	0.324103	0.104	0.00000000	0.00000000	0.00000000
## 40	0.31879	0.104	0.00000000	0.00000000	0.00000000
## 41	0.313477	0.104	0.00000000	0.00000000	0.00000000
## 42	0.308164	0.104	0.00000000	0.00000000	0.00000000
## 43	0.302851	0.104	0.00000000	0.00000000	0.00000000
## 44	0.297538	0.104	0.00000000	0.00000000	0.00000000
## 45	0.292225	0.104	0.00000000	0.00000000	0.00000000
## 46	0.286911	0.104	0.00000000	0.00000000	0.00000000
## 47	0.281598	0.104	0.00000000	0.00000000	0.00000000
## 48	0.276285	0.104	0.00000000	0.00000000	0.00000000
## 49	0.270972	0.104	0.00000000	0.00000000	0.00000000
## 50	0.265659	0.104	0.00000000	0.00000000	0.00000000
## 51	0.260346	0.104	0.00000000	0.00000000	0.00000000
## 52	0.255033	0.104	0.00000000	0.00000000	0.00000000
## 53	0.24972	0.104	0.00000000	0.00000000	0.00000000
## 54	0.244407	0.104	0.00000000	0.00000000	0.00000000
## 55	0.239094	0.104	0.00000000	0.00000000	0.00000000
## 56	0.233781	0.104	0.00000000	0.00000000	0.00000000
## 57	0.228468	0.104	0.00000000	0.00000000	0.00000000
## 58	0.223155	0.104	0.00000000	0.00000000	0.00000000
## 59	0.217841	0.104	0.00000000	0.00000000	0.00000000
## 60	0.212528	0.104	0.00000000	0.00000000	0.00000000
## 61	0.207215	0.104	0.00000000	0.00000000	0.00000000
## 62	0.201902	0.104	0.00000000	0.00000000	0.00000000
## 63	0.196589	0.104	0.00000000	0.00000000	0.00000000
## 64	0.191276	0.104	0.00000000	0.00000000	0.00000000
## 65	0.185963	0.104	0.00000000	0.00000000	0.00000000
## 66	0.18065	0.104	0.00000000	0.00000000	0.00000000
## 67	0.175337	0.104	0.00000000	0.00000000	0.00000000
## 68	0.170024	0.104	0.00000000	0.00000000	0.00000000
## 69	0.164711	0.104	0.00000000	0.00000000	0.00000000
## 70	0.159398	0.104	0.00000000	0.00000000	0.00000000
## 71	0.154085	0.104	0.00000000	0.00000000	0.00000000
## 72	0.148771	0.104	0.00000000	0.00000000	0.00000000
## 73	0.143458	0.104	0.00000000	0.00000000	0.00000000
## 74	0.138145	0.104	0.00000000	0.00000000	0.00000000
## 75	0.132832	0.104	0.00000000	0.00000000	0.00000000
## 76	0.127519	0.104	0.00000000	0.00000000	0.00000000
## 77	0.122206	0.104	0.00000000	0.00000000	0.00000000
## 78	0.116893	0.104	0.00000000	0.00000000	0.00000000
## 79	0.11158	0.104	0.00000000	0.00000000	0.00000000

```
## 80 0.106267 0.104 0.00000000 0.00000000 0.00000000
## 81 0.100954 0.104 0.00000000 0.00000000 0.00000000
## 82 0.095641 0.104 0.00000000 0.00000000 0.00000000
## 83 0.090328 0.104 0.00000000 0.00000000 0.00000000
## 84 0.085015 0.104 0.00000000 0.00000000 0.00000000
## 85 0.079701 0.104 0.00000000 0.00000000 0.00000000
## 86 0.074388 0.104 0.00000000 0.00000000 0.00000000
## 87 0.069075 0.104 0.00000000 0.00000000 0.00000000
## 88 0.063762 0.104 0.00000000 0.00000000 0.00000000
## 89 0.058449 0.104 0.00000000 0.00000000 0.00000000
## 90 0.053136 0.104 0.00000000 0.00000000 0.00000000
## 91 0.047823 0.104 0.00000000 0.00000000 0.00000000
## 92 0.04251 0.104 0.00000000 0.00000000 0.00000000
## 93 0.037197 0.104 0.00000000 0.00000000 0.00000000
## 94 0.031884 0.104 0.00000000 0.00000000 0.00000000
## 95 0.026571 0.104 0.00000000 0.00000000 0.00000000
## 96 0.021258 0.104 0.00000000 0.00000000 0.00000000
## 97 0.015944 0.104 0.00000000 0.00000000 0.00000000
## 98 0.010631 0.104 0.00000000 0.00000000 -0.008353296
## 99 0.005318 0.104 0.00000000 0.00000000 -0.029626882
## 100 5e-06 0.104 0.02037941 0.006272124 -0.052047903
```

```
# True estimates from GLM
```

```
#as.vector(glm(y ~ X[, -1], family = binomial)$coefficients)
```

```
# True estimates from GLMNET
```

```
fit <- glmnet(X, y, family = "binomial", standardize = FALSE, lambda = lambda_seq, thresh = 1e-10)
```

```
fit_result <- fit$beta %>% as.matrix() %>% t() %>% data.frame() %>% `rownames<-`( NULL )
```

```
fit_result %>% mutate(lambda = fit$lambda %>% round(digits = 6)) %>% relocate(lambda)
```

```
##      lambda V1      V2      V3      V4
## 1  0.526000 0 0.00000000 0.00000000 0.00000000
## 2  0.520687 0 0.00000000 0.00000000 0.00000000
## 3  0.515374 0 0.00000000 0.00000000 0.00000000
## 4  0.510061 0 0.00000000 0.00000000 0.00000000
## 5  0.504748 0 0.00000000 0.00000000 0.00000000
## 6  0.499435 0 0.00000000 0.00000000 0.00000000
## 7  0.494122 0 0.00000000 0.00000000 0.00000000
## 8  0.488808 0 0.00000000 0.00000000 0.00000000
## 9  0.483495 0 0.00000000 0.00000000 0.00000000
## 10 0.478182 0 0.00000000 0.00000000 0.00000000
## 11 0.472869 0 0.00000000 0.00000000 0.00000000
## 12 0.467556 0 0.00000000 0.00000000 0.00000000
## 13 0.462243 0 0.00000000 0.00000000 0.00000000
## 14 0.456930 0 0.00000000 0.00000000 0.00000000
## 15 0.451617 0 0.00000000 0.00000000 0.00000000
## 16 0.446304 0 0.00000000 0.00000000 0.00000000
## 17 0.440991 0 0.00000000 0.00000000 0.00000000
## 18 0.435678 0 0.00000000 0.00000000 0.00000000
## 19 0.430365 0 0.00000000 0.00000000 0.00000000
## 20 0.425052 0 0.00000000 0.00000000 0.00000000
## 21 0.419738 0 0.00000000 0.00000000 0.00000000
```

## 22	0.414425	0	0.00000000	0.00000000	0.00000000
## 23	0.409112	0	0.00000000	0.00000000	0.00000000
## 24	0.403799	0	0.00000000	0.00000000	0.00000000
## 25	0.398486	0	0.00000000	0.00000000	0.00000000
## 26	0.393173	0	0.00000000	0.00000000	0.00000000
## 27	0.387860	0	0.00000000	0.00000000	0.00000000
## 28	0.382547	0	0.00000000	0.00000000	0.00000000
## 29	0.377234	0	0.00000000	0.00000000	0.00000000
## 30	0.371921	0	0.00000000	0.00000000	0.00000000
## 31	0.366608	0	0.00000000	0.00000000	0.00000000
## 32	0.361295	0	0.00000000	0.00000000	0.00000000
## 33	0.355981	0	0.00000000	0.00000000	0.00000000
## 34	0.350668	0	0.00000000	0.00000000	0.00000000
## 35	0.345355	0	0.00000000	0.00000000	0.00000000
## 36	0.340042	0	0.00000000	0.00000000	0.00000000
## 37	0.334729	0	0.00000000	0.00000000	0.00000000
## 38	0.329416	0	0.00000000	0.00000000	0.00000000
## 39	0.324103	0	0.00000000	0.00000000	0.00000000
## 40	0.318790	0	0.00000000	0.00000000	0.00000000
## 41	0.313477	0	0.00000000	0.00000000	0.00000000
## 42	0.308164	0	0.00000000	0.00000000	0.00000000
## 43	0.302851	0	0.00000000	0.00000000	0.00000000
## 44	0.297538	0	0.00000000	0.00000000	0.00000000
## 45	0.292225	0	0.00000000	0.00000000	0.00000000
## 46	0.286911	0	0.00000000	0.00000000	0.00000000
## 47	0.281598	0	0.00000000	0.00000000	0.00000000
## 48	0.276285	0	0.00000000	0.00000000	0.00000000
## 49	0.270972	0	0.00000000	0.00000000	0.00000000
## 50	0.265659	0	0.00000000	0.00000000	0.00000000
## 51	0.260346	0	0.00000000	0.00000000	0.00000000
## 52	0.255033	0	0.00000000	0.00000000	0.00000000
## 53	0.249720	0	0.00000000	0.00000000	0.00000000
## 54	0.244407	0	0.00000000	0.00000000	0.00000000
## 55	0.239094	0	0.00000000	0.00000000	0.00000000
## 56	0.233781	0	0.00000000	0.00000000	0.00000000
## 57	0.228468	0	0.00000000	0.00000000	0.00000000
## 58	0.223155	0	0.00000000	0.00000000	0.00000000
## 59	0.217841	0	0.00000000	0.00000000	0.00000000
## 60	0.212528	0	0.00000000	0.00000000	0.00000000
## 61	0.207215	0	0.00000000	0.00000000	0.00000000
## 62	0.201902	0	0.00000000	0.00000000	0.00000000
## 63	0.196589	0	0.00000000	0.00000000	0.00000000
## 64	0.191276	0	0.00000000	0.00000000	0.00000000
## 65	0.185963	0	0.00000000	0.00000000	0.00000000
## 66	0.180650	0	0.00000000	0.00000000	0.00000000
## 67	0.175337	0	0.00000000	0.00000000	0.00000000
## 68	0.170024	0	0.00000000	0.00000000	0.00000000
## 69	0.164711	0	0.00000000	0.00000000	0.00000000
## 70	0.159398	0	0.00000000	0.00000000	0.00000000
## 71	0.154085	0	0.00000000	0.00000000	0.00000000
## 72	0.148771	0	0.00000000	0.00000000	0.00000000
## 73	0.143458	0	0.00000000	0.00000000	0.00000000
## 74	0.138145	0	0.00000000	0.00000000	0.00000000
## 75	0.132832	0	0.00000000	0.00000000	0.00000000



## 76	0.127519	0	0.00000000	0.00000000	0.00000000
## 77	0.122206	0	0.00000000	0.00000000	0.00000000
## 78	0.116893	0	0.00000000	0.00000000	0.00000000
## 79	0.111580	0	0.00000000	0.00000000	0.00000000
## 80	0.106267	0	0.00000000	0.00000000	0.00000000
## 81	0.100954	0	0.00000000	0.00000000	0.00000000
## 82	0.095641	0	0.00000000	0.00000000	0.00000000
## 83	0.090328	0	0.00000000	0.00000000	0.00000000
## 84	0.085015	0	0.00000000	0.00000000	0.00000000
## 85	0.079701	0	0.00000000	0.00000000	0.00000000
## 86	0.074388	0	0.00000000	0.00000000	0.00000000
## 87	0.069075	0	0.00000000	0.00000000	0.00000000
## 88	0.063762	0	0.00000000	0.00000000	0.00000000
## 89	0.058449	0	0.00000000	0.00000000	0.00000000
## 90	0.053136	0	0.00000000	0.00000000	0.00000000
## 91	0.047823	0	0.00000000	0.00000000	0.00000000
## 92	0.042510	0	0.00000000	0.00000000	0.00000000
## 93	0.037197	0	0.00000000	0.00000000	0.00000000
## 94	0.031884	0	0.00000000	0.00000000	0.00000000
## 95	0.026571	0	0.00000000	0.00000000	0.00000000
## 96	0.021258	0	0.00000000	0.00000000	0.00000000
## 97	0.015944	0	0.00000000	0.00000000	0.00000000
## 98	0.010631	0	0.00000000	0.00000000	-0.008376238
## 99	0.005318	0	0.00000000	0.00000000	-0.029715465
## 100	0.000005	0	0.02045443	0.006293078	-0.052235340